REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: AFIT/GI/NR 87-84T
2. GOVT ACCESSION NO.:
3. RECIPIENT'S CATALOG NUMBER: A186 127

4. TITLE (and Subtitle): On The Parabolized Navier-Stokes Equations Without Sublayer Assumptions
5. TYPE OF REPORT & PERIOD COVERED: THESIS/DISSERTATION
6. PERFORMING ORG. REPORT NUMBER:

7. AUTHOR(s): Stephen Christian Pluntze
8. CONTRACT OR GRANT NUMBER(s):

9. PERFORMING ORGANIZATION NAME AND ADDRESS: AFIT STUDENT AT: Massachusetts Institute of Technology
10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS:

11. CONTROLLING OFFICE NAME AND ADDRESS: AFIT/NR, WPAFB OH 45433-6583
12. REPORT DATE: 1986
13. NUMBER OF PAGES: 147

14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office):
15. SECURITY CLASS. (of this report): UNCLASSIFIED
15a. DECLASSIFICATION/DOWNGRADING SCHEDULE:

16. DISTRIBUTION STATEMENT (of this Report)

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES: APPROVED FOR PUBLIC RELEASE: IAW AFR 190-1

LYNN E. WOLAVER 23 Oct 87
Dean for Research and Professional Development
AFIT/NR

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)
ATTACHED

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

(1)

84

Enclosed is the Master of Science Thesis for Capt. Stephen C. Pluntze. It is 147 pages long and was submitted to the Dept of Aeronautical and Astronautical Engineering at the Massachusetts Institute of Technology for the degree of Master of Science in Aeronautical and Astronautical Engineering.

ON THE PARABOLIZED NAVIER - STOKES EQUATIONS WITHOUT SUBLAYER ASSUMPTIONS

by

STEPHEN CHRISTIAN PLUNTZE
Captain, United States Air Force

BS, UNITED STATES AIR FORCE ACADEMY
(1982)

SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE
DEGREE OF

MASTER OF SCIENCE IN
AERONAUTICS AND ASTRONAUTICS

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

January, 1987

Signature of Author _____
Department of Aeronautics and Astronautics
January 23, 1987

Certified by _____
Prof. Judson R. Baron, Thesis Supervisor
Department of Aeronautics and Astronautics

Accepted by _____
Prof. Harold Y. Wachman
Chairman, Department Graduate Committee

ON THE PARABOLIZED NAVIER-STOKES EQUATIONS WITHOUT SUBLAYER ASSUMPTIONS

by

STEPHEN CHRISTIAN PLUNTZE
Captain, United States Air Force

Submitted to the Department of Aeronautical Engineering
on January 23, 1986 in partial fulfillment of the requirements
for the Degree of Master of Science
in Aeronautical Engineering.

ABSTRACT

A new implicit, iterative method of solving the Parabolized Navier-Stokes (PNS) Equations claims to overcome the elliptic character of the embedded subsonic sublayer by explicitly introducing pressure as an additional state variable. The Bhutta-Lewis approach makes no sublayer pressure assumptions. The validity and basis of that method is explored in this thesis by examining the relevant eigenvalues governing marching stability. An original code was also developed in order to examine the numerical character of the marching, iterative solutions as they develop. Test cases were carried out for a two dimensional wedge configuration at Mach numbers 3 and 15 and Reynolds numbers ranging from 4×10^3 to 1×10^7 at the initial data plane.

An eigenvalue analysis disclosed that the method is unstable in subsonic regions. Introducing the additional state variable does not change the character of the equations.

Results for the test cases confirmed the presence of instability. Classic departure behavior was produced in tightly clustered grids and convergence to separated flow was shown in less clustered grids. Marching was achieved only in relatively high Reynolds number flow with a large stable marching step size. Uniform step size was used in this study, but it is possible that variable step sizes allowed Bhutta and Lewis to march successfully; however, no discussion of the step size variation and its relationship to stability appeared in their original work.

Thesis Advisor: Dr. Judson R. Baron
Title: Professor of Aeronautics and Astronautics

ACKNOWLEDGEMENTS

This work has not been a solo effort by any means. Many people have contributed to its completion. My thesis advisor, Professor Judson R. Baron, taught me what it really means to be a good analytical Computational Fluid Dynamicist. He always provided new and revealing insights into the problem. I am deeply indebted for his guiding hand.

Special thanks goes to the MIT Lincoln Laboratory for the use of their computer. Specifically, my gratitude extends to Dr Michael Judd, Mr. Craig Perini, Mr. Jack Kelly, Mr. Harvey Fenton, and especially Mr. Jim Schrock, who wrote the graphics software and helped me use it. I couldn't have completed this thesis without their help and advice.

The Education Office personnel at the C.S. Draper Laboratory were extremely helpful - not only with office space and word processing equipment, but with friendship and support. I will always have fond memories of Dr. David Burke, Mrs. Vilma Dunham, Miss Shirley Grady and my typist, Mrs. Peg Hood, who was truly a Godsend.

Most importantly, I want to thank my wife, Brenda. She not only provided love and support when I was too busy to spend time with her, but she presented us with our first child during our stay in Boston. She is wonderful.

My great thanks to you all.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF FIGURES (continued)

LIST OF FIGURES (continued)

LIST OF TABLES

LIST OF VARIABLES

A, B, C	Block Tridiagonal Matrices
G	Right Hand Side Vector of Tridiagonal Form
J	Grid Transformation Jacobian
Pr	Prandtl Number
Re_L	Reynolds Number Based on Reference Length, L
X	Body Surface Coordinate
Z	Surface Normal Coordinate
A_1, A_2, M	Jacobian Matrices of Flux Vectors
A_θ	Jacobian Matrix of State Equation Vector
F_1, F_2, S	Flux Vectors
H	State Equation Vector
j	Streamwise Node
L	Vertical Node
q, Q	Smoothed and Unsmoothed State Vector
s	Grid Clustering Parameter
ξ_x, ξ_z, ζ_x, ζ_z	Metrics
Δx	Streamwise Physical Stepsize
$\Delta\zeta$ $\Delta\xi$	Change in Vertical and Horizontal Computational Plane.
θ	State Equation Parameter
ϵ	Nondimensionalization Parameter = M_∞/Re_L
β	Shock Angle
δ_s	Subsonic Layer Height
ω	Smoothing Parameter
δ^*	Boundary Layer Displacement Thickness

1.0 INTRODUCTION

Bhutta and Lewis[1] have proposed a new and strikingly different method of solving the Parabolized Navier-Stokes (PNS) equations. This new method claims greater accuracy for the same computing time. In essence, the original idea behind the new scheme is to solve the normal system of non-linear partial differential equations with the equation of state included as an additional equation and pressure as an additional state variable.

The entire method will be reconstructed and reexamined in detail here. The primary focus will be on the classical substitution for pressure in the momentum and energy equations, and the pressure role in Reference 1 where it is kept separate in the state equation. Specifically, a conventional two-dimensional state vector of the form:

$$q = \begin{pmatrix} \rho \\ \rho u \\ \rho w \\ \rho E \end{pmatrix}$$

is augmented in the new PNS scheme[1] and appears as

$$q = \begin{pmatrix} \rho \\ \rho u \\ \rho w \\ \rho T \\ p \end{pmatrix}$$

Although equations and unkowns consistently increase by one, two of the resulting state elements are equivalent according to the state equation. The equation system is of a mixed differential/algebraic type. Moreover, Bhutta and Lewis

explicitly rule out the need for a sublayer assumption. Previous PNS codes have introduced pressure assumptions in the subsonic portion of the boundary layer in order to eliminate elliptical constraints on marching downstream and the development of departure solutions.

The PNS Equations were developed to save on the large storage requirements needed for the full Navier-Stokes equations. Their development arose from the need to solve large numbers of problems where viscous contributions are dominant in the direction normal to the streamwise direction. In other words, flows that are boundary-layer-like are suitable to be solved by the PNS equations. This includes a large class of high speed flows which are of current interest.

Previous PNS codes have used some sort of sublayer pressure assumption to enable a marching solution in the streamwise direction. Lin and Rubin[14] have used pressure from experiments and have set the subsonic streamwise pressure derivative equal to the derivative at the edge of the subsonic layer. Lubard and Helliwell[3,14] use a backward difference for the streamwise pressure derivative. Vigneron et al[3,14] treated the pressure derivative exactly in the supersonic region defined by

$$M^2 > 1/(2-\gamma)$$

For other Mach numbers the pressure derivative is suppressed by the factor

$$[1/(\gamma-1)][1-1/(1+(\gamma-1)M^2)]$$

Schiff and Steger[6] specified the subsonic pressure to be equal to pressure at the bottom of the supersonic region. Kaul[14] uses a global relaxation over the entire domain while still maintaining the marching scheme. Lighthill[13] in 1953 found that departure, or separation-like solutions, are observed in the boundary layer equations when the streamwise pressure gradient is not specified. Similar behavior has been investigated in the PNS equations by Barnett.[5]

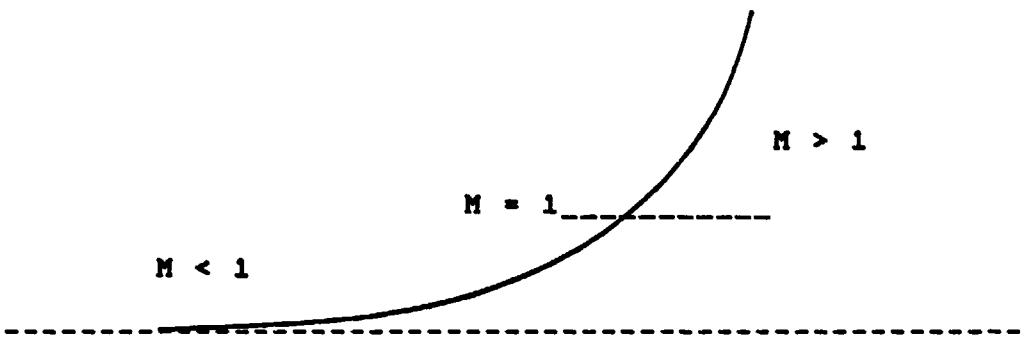


Figure 1.1 Supersonic Velocity Profile

Since the flow within some region near a surface is always subsonic (Fig. 1.1) and hence elliptic, those assumptions have been required to change the physical description from a mixed hyperbolic/elliptic to hyperbolic character. Reference 1 on the other hand, claims that without changing the description of the physical nature of the flowfield, i.e., without sublayer assumptions, the code can be marched in hyperbolic fashion. Justification rests there upon a stability analysis which is

based on a modified state equation. Further analysis will also be presented in the stability chapter of this thesis.

Although cast in the framework of a PNS problem, the implications of such modification of an equation system to achieve stable marching would impact in many areas besides fluid dynamics. Being able to mathematically change the character of a system of equations without changing the physical aspects of particular problems would benefit most if not all of the physical sciences with promises of greater accuracy and efficiency of solutions. It is therefore of some importance to rigorously examine the suggested new method.

2.0 PROBLEM DESCRIPTION

In reproducing the new PNS scheme, the emphasis here is on the role of the individual procedural components of the algorithm: normalization, initial conditions, step size, iterative technique, matrix formulations, smoothing, etc. and how these relate to convergence and marching. This study is concerned more with the evolving solution rather than the speed, accuracy, or efficiency of the scheme.

In order to understand how the method works, the first objective of this thesis was to devise a faithful code which would provide sample calculations for a simple but meaningful physical problem. A wedge flow at M=3 and 15 and Reynolds numbers ranging from 4×10^3 to 1×10^7 are used. The second purpose involves stability questions[1,9,13,14,15] and includes eigenvalue analyses of a previous[8] scheme and the new PNS scheme, as well as researching the behavior of systems of partial differential equations.

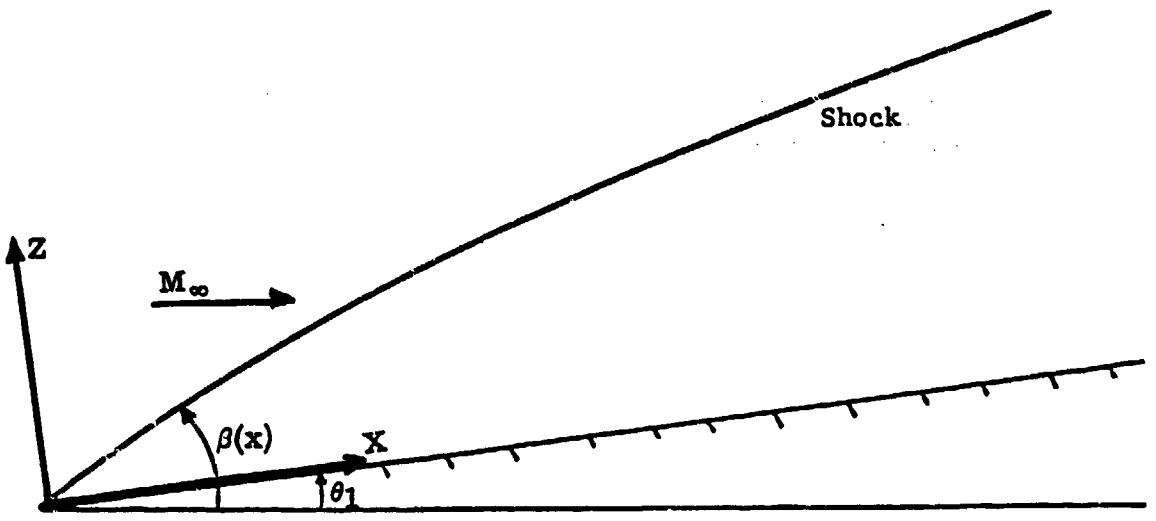
2.1 PROBLEM AS TESTED

Reference 1 used the new PNS scheme on a blunt body at Mach 25 at Reynolds numbers of 2.92×10^3 and 1.72×10^3 based on nose radius. The reason for these values was to test the scheme at high velocities and low Reynolds Number with large viscous

effects, where conventional PNS schemes break down.

Since the current study is concerned primarily with the validity of the concept in question, it uses a simpler problem: a simple wedge flying at Mach 3 and 15 and Reynolds numbers from 4×10^3 to 1×10^7. Reynolds number can be varied to simulate more or less viscous conditions.

The figure below shows the coordinate system of the wedge and Table 2.1 shows the freestream conditions.



θ_1 = wedge angle
β = shock angle

Figure 2.1 Wedge Coordinate System

FREESTREAM VALUES* 80,000 FEET		
DENSITY	8.6×10^{-8}	SL/FT
PRESSURE	58.125	LB/FT^2
TEMPERATURE	389.99	°R
SONIC SPEED	977.6	FT/SEC
VISCOSITY	9.7×10^{-6}	SL/FTSEC

Table 2.1 Freestream Values

3.0 CONSERVATION EQUATIONS

As the name suggests, the Parabolized Navier-Stokes Equations are derived from the full Navier-Stokes Equations. In a somewhat similar approximation as for the boundary layer equations, only some of the viscous terms are retained. The PNS system is applicable to supersonic flow and, since it is valid in both the viscous and inviscid portions of the flow, the interaction between such regions is included automatically.[3]

In a normal supersonic viscous flow, the no-slip condition at the surface implies that the boundary layer is at some point subsonic and transitions to supersonic flow within the boundary layer (Figure 1.1).

A consequence of this subsonic sublayer is that the streamwise flow is elliptic despite the outer supersonic flow and the solution cannot be marched downstream in a hyperbolic/ parabolic fashion. The constraint introduced by sublayer assumptions is control of the upstream communication of pressure disturbances through the subsonic region. By making appropriate pressure assumptions, such as constant pressure across the sublayer, the solution can be marched in the streamwise direction. This has been demonstrated in a number of previous PNS schemes.[3,8,9,14]

The novelty of the current method is that no such assumption is made. The claim is that by treating the equation of state as a separate but coupled equation, the elliptic nature of the problem is circumvented. Although no assumptions may mean greater accuracy, it is necessary to demonstrate that the state equation treatment somehow alters the character of the equation system. This point will be considered further.

3.1 DERIVATION

In two dimensions, the Navier-Stokes Equations in vector form appear as:

$$q_{,t} + [Ei - Ev]_{,x} + [Gi - Gv]_{,z} - H = 0 \qquad 3.1$$

where q is the state vector, the comma indicates partial differentiation and E and G are the X and Z flux vectors. The i and v refer to inviscid and viscous and all variables are dimensional. The vector H contains the state equation.

The X and Z directions correspond to the components along and normal to the wedge surface of the current problem (Fig. 2.1).

Here:

$$q = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho T \\ P \end{pmatrix} \qquad 3.2$$

P is the extra state variable corresponding to the added equation of state. The density-temperature product has been used here for consistency with the convention in Reference 1.

The derivation of PNS equations involves an order of magnitude analysis, much like the boundary layer equations. The most common form (Lubard and Helliwell, 1973, 1974)[3] is obtained by assuming steady flow and that the streamwise viscous derivative terms are negligible compared to the normal (and transverse, if 3-D) viscous derivative terms. In other words, the PNS equations are derived simply by dropping all viscous terms containing partial derivatives with respect to the streamwise direction. With this in mind, it is rather simple to reduce equations 3.1 to PNS form:

$$Ei_{,x} + Gi_{,z} = Gv_{,z} + H \qquad 3.3$$

The state vector remains the same as in equation 3.2. Since the equation of state is a separate equation and is algebraic in form, it is the sole contributor to the separate vector, H.

The individual flux vectors are:

$$Ei = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ (T/(\gamma-1) + V^2/2)\rho u \\ 0 \end{pmatrix}$$

where $V^2 = u^2 + w^2$

$$Gi = \begin{pmatrix} \rho w \\ \rho uv \\ \rho w^2 + p \\ (T/(\gamma-1) + V^2/2)\rho w \\ 0 \end{pmatrix} \qquad 3.4$$

$$Gv = \begin{pmatrix} 0 \\ \tau_{xz} \\ \tau_{zz} \\ u\tau_{xz} + w\tau_{zz} - q_z \\ 0 \end{pmatrix}$$

$$H = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ P - \rho RT \end{pmatrix}$$

The above flux vectors remain dimensional.

3.2 NONDIMENSIONALIZATION

Following the convention of Reference 1, the non-dimensional variables become:

$u = u'/a'_\infty$ (' now indicates a dimensional quantity)

$w = w'/a'_\infty$

$\rho = \rho'/\rho'_\infty$

$T = T'/T'_\infty$

$P = P'/(\rho_\infty a_\infty^2)'$

$\mu = \mu'/\mu'_\infty$

$X = X'/L'$

$Z = Z'/L'$

When these definitions are substituted into the dimensional equations 3.4, the result is the nondimensional PNS equations:

$$E_{i,x} + G_{i,z} = \epsilon G_{v,z} + H \qquad 3.5$$

The nondimensional state equation contained in the H vector is $p = \rho T$. Unless stated otherwise, all equations and parameters henceforth will be non-dimensional.

The factor ϵ results from the normalization process and is defined as

$$\epsilon = M_e/Re_L \qquad 3.6$$

Re_L is the Reynolds number based on the reference length, L'.

$$Re_L = (\rho_e u_e L/\mu_e)' \qquad 3.7$$

For the viscous wedge flow in this study a characteristic scale parameter is implied by the boundary layer displacement thickness, δ^*. In dimensional form[3]

$$\delta^{*\prime} = (1.7208)X'/(Re_L)^{1/2} \qquad 3.8$$

Squaring both sides, multiplying both sides by $X'/(L')^2$ and simplifying gives (in nondimensional form)

$$(\delta^*)^2 = (2.96)X/Re_L$$

L' is the reference length chosen to be the wedge surface location at which $X=(X'/L')=1$, so that Re_L follows from:

$$Re_{\delta^*} = 2.96/(\delta^*)^2 \qquad 3.9$$

δ^* is the nondimensional boundary layer displacement thickness at X=1. This is estimated by knowing the boundary layer thickness at the starting point. If the starting plane is at X=1 there is a corresponding Re_{δ^*} and L' for any choice of δ^* and vice versa. For example, at Me = 3 and with a 7° (half angle) wedge, the shock angle is about 24°. From Figure 2.1, with X = 1, Z=tan(shock angle - wedge angle) or Z = tan (24° - 7°) = 0.3. Assuming a boundary layer that occupies 10% of the shock layer, its thickness will be about 0.03 and a reasonable approximation for δ^* might be 0.01. Using 3.9:

$$Re_{\delta^*} = 2.96 \times 10^4$$

and using 3.7

$$L' = 3.45 \times 10^{-8} \text{ ft} \qquad 3.10$$

Of course increasing or decreasing the boundary layer thickness will result in changes in the estimate of δ^* and the viscous nature of the problem for the corresponding smaller or larger Reynolds number.

3.3 GENERAL COORDINATE TRANSFORMATION

The PNS equations in the form of 3.5 are appropriate if the grid is everywhere orthogonal and rectangular. Unfortunately, a match of grid with the wedge and shock boundaries requires a transformation. More generally the equations must be prepared to allow a general coordinate transformation from a nonorthogonal physical plane to an orthogonal computational plane. The details of a specific grid transformation for the wedge will be explained in Chapter 4; the general transformation of the PNS equations will be discussed here.

In essence, the goal is to relate the non-orthogonal X, Z system to an orthogonal ξ, ζ system. We take ξ as the streamwise direction, and ζ as the crossflow direction. To begin, start again with the full non-dimensional steady Navier-Stokes equations:

$$[Ei - \epsilon Ev]_{,x} + [Gi - \epsilon Gv]_{,z} - H = 0 \qquad 3.11$$

To transform to the ξ, ζ system, the derivatives $(,X)$ and $(,Z)$ must be changed to a combination of the derivatives $(,\xi)$ and $(,\zeta)$. By the chain rule,

$$
\begin{aligned}
(,X) &= (\xi_{,x})(,\xi) + (\zeta_{,x})(,\zeta) \\
&= \xi_x(,\xi) + \zeta_x(,\zeta) \qquad 3.12 \\
(,Z) &= \xi_z(,\xi) + \zeta_z(,\zeta)
\end{aligned}
$$

This assumes a most general transformation of the form:

$$\xi = \xi(X,Z)$$

$$\zeta = \zeta(X,Z)$$

Now use 3.12 in 3.11 to get

$$
\begin{aligned}
&[\xi_x(Ei-\epsilon Gi) + \xi_z(Ev-\epsilon Gv)],\xi \\
&+ [\zeta_x(Ei-\epsilon Gi) + \zeta_z(Ev-\epsilon Gv)],\zeta = 0 \qquad 3.13
\end{aligned}
$$

In keeping with the assumptions to derive the PNS equations from the Navier-Stokes Equations, the streamwise viscous derivatives are omitted, which leaves:

$$
\begin{aligned}
&[\xi_x(Ei-\epsilon Gi)],\xi \\
&+ [\zeta_x(Ei-\epsilon Gi) + \zeta_z(Ev-\epsilon Gv)],\zeta \qquad 3.13a
\end{aligned}
$$

The Ev and Gv viscous vectors contain X and Z derivatives which must also be transformed to ξ and ζ. The entire derivation may be found in Appendix A. The transformed PNS equations become:

$$F_1,\xi + F_2,\zeta = \epsilon S,\zeta + H \qquad 3.14$$

Here the notation is different to emphasize that the transformation has been completed and H is still the vector

containing the equation of state terms. The H vector is not transformed because of its algebraic nature. Other vectors are:

$$F_1 = (1/J)\begin{pmatrix} \rho U_1 \\ \rho u U_1 + \xi_x p \\ \rho w U_1 + \xi_z p \\ (T/(\gamma-1)+V^2/2)\rho U_1 \\ 0 \end{pmatrix}$$

$$F_2 = (1/J)\begin{pmatrix} \rho U_2 \\ \rho u U_2 + \zeta_x p \\ \rho w U_2 + \zeta_z p \\ (T/(\gamma-1)+V^2/2)\rho U_2 \\ 0 \end{pmatrix} \qquad 3.15$$

$$S = (\mu/J)\begin{pmatrix} 0 \\ M_0 u,\zeta + (M_{x\,x} u,\zeta + M_{x\,z} w,\zeta)/3 \\ M_0 w,\zeta + (M_{x\,z} u,\zeta + M_{z\,z} w,\zeta)/3 \\ M_0 (T,\zeta/(Pr(\gamma-1)) + uu,\zeta + ww,\zeta) \\ \quad + (M_{x\,x} uu,\zeta + M_{z\,z} ww,\zeta \\ \quad + M_{x\,z} (wu,\zeta + uw,\zeta))/3 \\ 0 \end{pmatrix}$$

And:

$$U_1 = \xi_x u + \xi_z w$$

$$U_2 = \zeta_x u + \zeta_z w$$

$$M_{x\,x} = \zeta_x^2$$

$$M_{x\,z} = \zeta_x \zeta_z$$

$$M_{z\,z} = \zeta_z^2$$

$M_\theta = M_{x\,x} + M_{z\,z}$

$V^2 = u^2 + w_2$

J = Transformation Jacobian defined in Eq. (4.7)

The Jacobian is included to make the entire transformation conservative.[3] This will be seen more clearly in Chapter 4.

Equations 3.14 are to be solved. However, in preparation for a later stability analysis, a differential version of the equation of state introduced in Reference 1 must be noted:

$$\theta(P,\xi + P,\zeta) + \gamma p = \rho T$$

Here θ is a "small" number. With this version of the equation of state the final versions of F_1 and F_2 become:

$$F_1 = (1/J)\begin{pmatrix} \rho U_1 \\ \rho u U_1 + \xi_x p \\ \rho w U_1 + \xi_z p \\ (T/(\gamma-1)+V^2/2)\rho U_1 \\ \theta p \end{pmatrix} \qquad 3.16$$

$$F_2 = (1/J)\begin{pmatrix} \rho U_2 \\ \rho u U_2 + \zeta_x p \\ \rho w U_2 + \zeta_z p \\ (T/(\gamma-1)+V^2/2)\rho U_2 \\ \theta p \end{pmatrix}$$

4.0 GRID AND GEOMETRY ANALYSIS

The viscous wedge problem clearly involves a boundary layer. The presence of viscosity implies that the location of the shock is unknown beforehand, unlike the inviscid problem. The shock locus is found as the solution is marched along the wedge. For a shock fitting approach, the shock serves as the upper boundary of the described domain. A convenient grid is one that conforms to the physical boundaries.

In order to solve the flow field numerically consider a rectangular grid in a computational plane. The relationship, or transformation, between the physical plane and the computational plane provides the metrics that were developed in Chapter 3.0, equations 3.12:

$$\xi_x, \xi_z, \zeta_x, \zeta_z$$

To perform the metrics analysis, begin with a general relation between the physical and computational planes. This general relation has the form:

$$\xi = \xi(X, Z) \qquad 4.1$$

$$\zeta = \zeta(X, Z) \qquad 4.2$$

and serves as a mapping between one plane and the other.

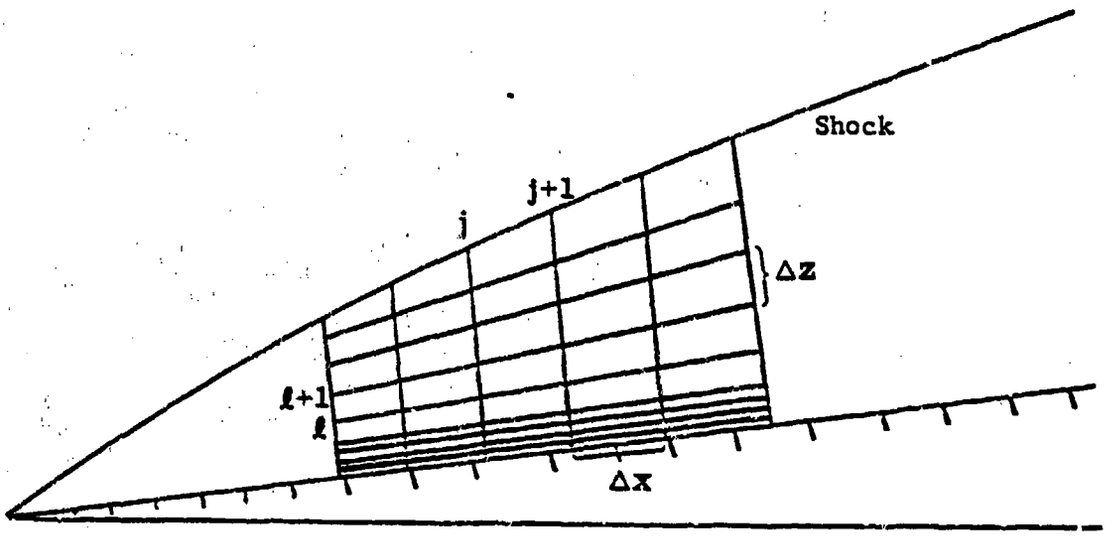


Figure 4.1 Physical Plane

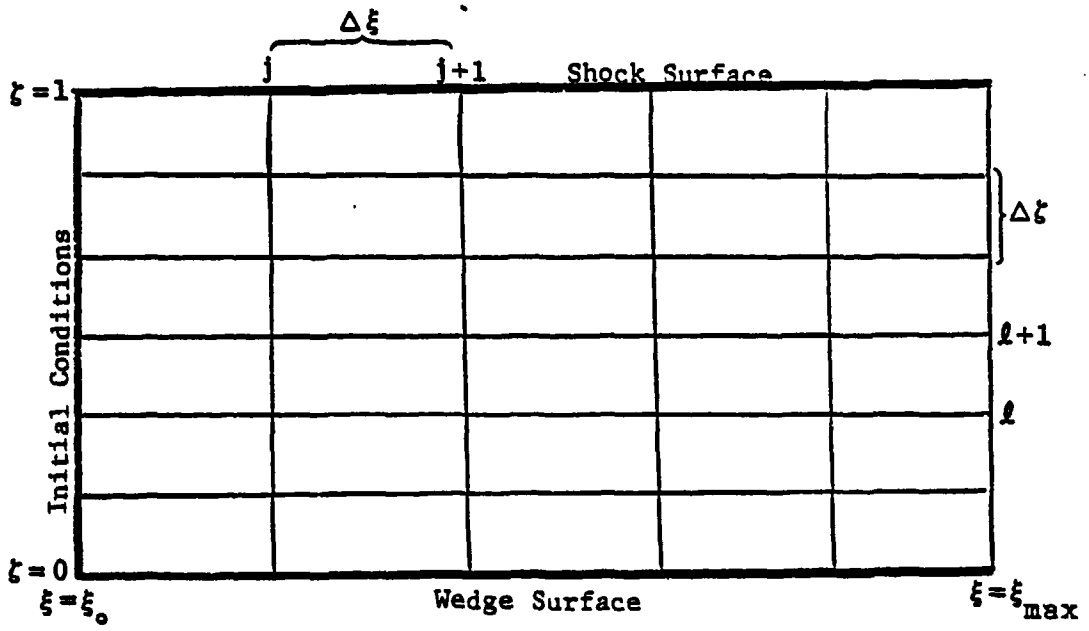


Figure 4.2 Computational Plane

From the chain rule,

$$(,\xi) = (X,\xi)(,x) + (Z,\xi)(,z) \qquad 4.3$$

$$(,\zeta) = (X,\zeta)(,x) + (Z,\zeta)(,z) \qquad 4.4$$

Since the transformation is from the physical to the computational planes, the derivatives $(,x)$ and $(,z)$ must be written in terms of the derivatives $(,\xi)$ and $(,\zeta)$. Solving 4.3 and 4.4 simultaneously gives:

$$(,x) = J[(Z,\zeta)(,\xi) - (Z,\xi)(,\zeta)] \qquad 4.5$$

$$(,z) = J[(X,\xi)(,\zeta) - (X,\zeta)(,\xi)] \qquad 4.6$$

where the Jacobian of the transformation, J, is represented by

$$J = 1/[(X,\xi)(Z,\zeta) - (X,\zeta)(Z,\xi)] \qquad 4.7$$

and appears in equations 3.15 and 3.16.

Now the general transformation of Chapter 3 can be completed. By comparing 4.5 and 4.6 to equations 3.12 it is seen that the required metrics in terms of the physical plane are:

$$\xi_x = Z,\zeta(J) \qquad 4.8$$

$$\xi_x = -X_\zeta(J) \qquad 4.9$$

$$\zeta_x = -Z_\xi(J) \qquad 4.10$$

$$\zeta_z = X_\xi(J) \qquad 4.11$$

Since the physical geometry is nonuniform while the computational plane remains constant, these metrics must be recalculated at each vertical (L) node at all streamwise (j) locations. See Figures 4.1 and 4.2.

The final contribution to the grid transformation relates to fixing $\Delta\xi$ and $\Delta\zeta$ since they are used to find derivatives in the computational domain. At each streamwise station there are always the same number of points vertically, say LMAX. Since ζ goes from 0 to 1, (Fig. 4.2).

$$\Delta\zeta = 1/(LMAX-1)$$

The range of Z-coordinates in the physical plane vary with shock layer thickness, but ζ is always mapped to the region 0 to 1 with equally spaced points $\Delta\zeta$ apart.

The ξ coordinate goes from ξ_0 to ξ_{max}, dependent on the start and end of the physical problem and the transformation. Therefore:

$$\Delta\xi = (\xi_{max} - \xi_0)/(JMAX-1)$$

Where JMAX is the number of streamwise stations, the initial data station being station number 1.

As more node points are added vertically and horizontally, the accuracy improves according to the accuracy of the governing finite difference equations (Chapter 5). This is in contrast to Reference 1 in which the computational grid appears to hold $\Delta\xi$ and $\Delta\zeta$ constant at 1.0.

It remains to calculate the metrics using 4.8 - 4.11. One could do so numerically by taking differences. For example, (Equ. 4.10:

$$\zeta_x = -\Delta x J/\Delta\xi$$

$$= -J(Z_{J+1,L} - Z_{J,L})/\Delta\xi$$

Where J must also be found by differencing. Of course knowing the physical grid locations at the j + 1 station in order to do the differencing requires knowledge of the shock location. The

shock prediction will be discussed in the chapter on boundary conditions.

If one knows the specific correspondence between the physical and computational planes, as in equations 4.1 and 4.2, the metrics may be found explicitly. For the wedge problem, the grid transformation is given by:

$$\xi = X \qquad 4.12$$

$$\zeta = (1/a)\sinh^{-1}(\sinh(a)[Z/Z_{s_{WALL}}]) \qquad 4.13$$

Equation 4.13 was chosen to cluster the grid points in the boundary layer[4]. With an "a" value of 3.0 for example, approximately 40% of the grid points lie within the boundary layer. The "a" value can be varied to give the desired grid clustering. Equation 4.13 can be rewritten as:

$$Z = (\sinh(a\zeta)/\sinh(a))[Z_{s_W}(x)] \qquad 4.14$$

Where Z_{s_W} is a function of X and hence (by 4.12), of ξ also. By 4.12, 4.13 and 4.14:

$$X,\xi = 1$$

$$X,\zeta = 0$$

$$Z,\xi = (\sinh(a\zeta)/\sinh(a))[Z'_{s_W}]$$

$$Z,\zeta = (a)(Z_{s_W})(\cosh(a\zeta)/\sinh(a))$$

where subscript SH refers to the value at the shock and

$$Z'_{SH} = (Z_{J+1} - Z_J)_{SH} / \Delta\xi$$

From the above formulae and equations 4.8 - 4.11;

f_1, f_2, f_3, and f_4

can be constructed at each streamwise station and at each vertical node as the solution advances in the streamwise direction.

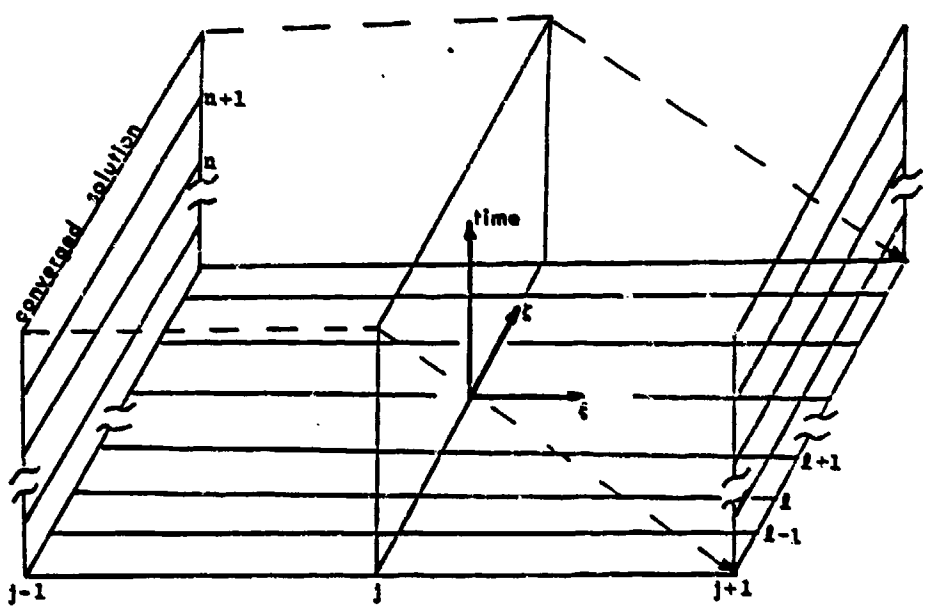
5.0 FINITE DIFFERENCE ALGORITHM

As we have seen, the non-dimensional 2-D PNS equations can be written as:

$$F_{1,\xi} + F_{2,\zeta} = \epsilon S_{,\zeta} + H \qquad 5.1$$

where F_1, F_2, S, and H are vectors.

The new scheme is implicit and iterative which means that the solution is moved forward in space from index j, say, to j+1 and then iterated in "pseudo time" from n to n+1 until it converges (Fig 5.1). The term "pseudo time" refers to the fact that only the converged steady state solution has physical meaning.



n=time index L=vertical index j=horizontal index

Figure 5.1 Index Notation

With reference to Figure 5.1, the converged solution at steps j and j-1 are used as the starting solution for step j+1. At step j+1, the solution is then iterated until convergence. At step j+1 we have:

$$(F_1,_\xi + F_2,_\zeta = S,_\zeta + H)_L{}^{j+1,n+1} \qquad 5.2$$

Since the method is implicit, all vertical L-node grid values at the n+1 level are found simultaneously in contrast to an explicit scheme which would find the n+1 values one at a time from the already known n values.

The above equation is not useful since the solution at n+1 is unknown -- the n+1 level must be tied to the n level. If we assume that the solution at level n+1 is close to that of the n-th level[1], Taylor Series expansions in pseudo time for each term in 5.2 gives:

$$F_1{}^{j+1,n+1} = F_1{}^{j+1,n} + (A_1{}^{j+1,n}\Delta q^{n+1}) \qquad 5.3$$

$$F_2{}^{j+1,n+1} = F_2{}^{j+1,n} + (A_2{}^{j+1,n}\Delta q^{n+1}) \qquad 5.4$$

$$S^{j+1,n+1} = S^{j+1,n} + (M^{j+1,n}\Delta q^{n+1}) \qquad 5.5$$

$$H^{j+1,n+1} = H^{j+1,n} + (A_0{}^{j+1,n}\Delta q^{n+1}) \qquad 5.6$$

Where: $\Delta q^{n+1} = q^{j+1,n+1} - q^{j+1,n}$ and these expansions are valid at each vertical node.

A_1, A_2, A_3, and M are called the Jacobian matrices, not to be confused with the transformation Jacobian (Chapter 4). They are 5 x 5 matrices since there are five elements in the state vector. The Jacobian matrices are formed by taking the partial derivatives of the flux vectors with respect to the state vector. The elements of the Jacobian matrices are given in Appendix B.

Substituting equations 5.3 - 5.6 into 5.2 gives:

$$[F_1^{j+1,n} + A_1^{j+1,n}\Delta q^{n+1}],_\xi + [F_2^{j+1,n} + A_2^{j+1,n}\Delta q^{n+1}],_\zeta = \epsilon[S^{j+1,n} + M^{j+1,n}\Delta q^{n+1}],_\zeta + [H^{j+1,n} + A_3^{j+1,n}\Delta q^{n+1}] \qquad 5.7$$

The streamwise derivative was given special treatment in Reference 1 so that the truncation error would be

$$O(\Delta q^{n+1})^2$$

instead of

$$O(\Delta q^{j+1})^2$$

as in conventional PNS schemes.[8,9,13,14,15] It is preferable to have temporal errors since they vanish for the converged solution.

The streamwise derivative from 5.2 becomes:

$$(F_i,\xi)^{j+1,n+1} = (F_i{}^{j+1,n+1} - F_i{}^{j})/\Delta\xi + O(\Delta\xi)$$

$$=[A_i{}^{j+1,n}\Delta q^{n+1}+(F_i{}^{j+1,n}-F_i{}^{j})]/\Delta\xi + O(\Delta\xi,(\Delta q^{n+1})^2)$$

$$=A_i{}^{j+1,n}\Delta q^{n+1}/\Delta\xi + (F_i,\xi)^{j+1,n}$$

With this formulation of the streamwise derivative, equation 5.7 appears as:

$$[(A_i/\Delta\xi - A_0)(\Delta^{n+1}) + (A_2-\epsilon M),\zeta(\Delta q^{n+1})]_L{}^{j+1,n} = -[F_i,\xi + F_2,\zeta - \epsilon S,\zeta - H]_L{}^{j+1,n} = G_L{}^{j+1,n}\text{, say} \qquad 5.8$$

Equations 5.8 and 5.7 are equivalent but 5.7 has been changed to explicitly show how the streamwise differencing is performed. Equation 5.8 is accurate to $O(\Delta q^{n+1})^2$ in pseudo time and is conservative in the limit of convergence.

It is convenient to call the right hand side of equation 5.8 $G_L{}^{j+1,n}$ for ease of notation. It is seen that on the right side all values are in terms of the n-th level, which are known. On the left, the coefficients of Δq^{n+1} are also from the n-th level. The problem is elliptic in the ζ direction so central differences are used for vertical

differencing. Upon using a central difference on the left of 5.8, the final form becomes:

$$(A_1/\Delta\xi - A_0)_L{}^{j+1,n}(\Delta q^{n+1})$$
$$+ \{((A_2-\epsilon M)_{L+1}/(2\Delta\zeta))(\Delta q^{n+1})$$
$$- ((A_2-\epsilon M)_{L-1}/(2\Delta\zeta))(\Delta q^{n+1})\}^{j+1,n}$$
$$= G_L{}^{j+1,n} \qquad 5.9$$

Looking at 5.9, the block tridiagonal form begins to appear. On the right, there is the vector $G^{j+1,n}$ at all vertical L nodes. On the left, there are three matrices; one at node L, one at node L+1 and the other at node L-1. The equation solves for the vectors $\Delta q_L{}^{n+1}$. Once $\Delta q_L{}^{n+1}$ has been found with a block tridiagonal solver, the new state vector can be found from:

$$q_L{}^{j+1,n+1} = q_L{}^{j+1,n} + \Delta q_L{}^{j+1,n+1} \qquad 5.10$$

Equations 5.9 and 5.10 do not address smoothing considerations, which will be added in Chapter 7.

To illustrate the block tridiagonal construct a bit more clearly, set

$$A = -[(A_2 - \epsilon M)/(2\Delta\zeta)]_{L-1}^{j+1,n}$$

$$C = [(A_2 - \epsilon M)/(2\Delta\zeta)]_{L+1}^{j+1,n} \qquad 5.11$$

$$B = (A_1/\Delta\xi - A_0)_L^{j+1,n}$$

Then the block tridiagonal equation appears as:

$$\begin{bmatrix} BC & & & \\ ABC & & & \\ & \ddots & & \\ & & ABC & \\ & & & AB \end{bmatrix} \times \begin{pmatrix} \Delta q_{L=2} \\ \vdots \\ \Delta q_{L=LMAX-1} \end{pmatrix} = \begin{pmatrix} G_{L=2} \\ \vdots \\ G_{L=LMAX-1} \end{pmatrix} \qquad 5.12$$

Boundary conditions for the matrix operations will be explained in the next Chapter.

A remaining question is how to find the right side of 5.8; namely, the derivatives:

$$F_1,_\xi \qquad F_2,_\zeta \qquad S,_\zeta$$

As mentioned earlier, the ζ derivatives are modeled by central differencing, i.e.,

$$F_{\xi}, \zeta = (F_{\xi L+1} - F_{\xi L-1})^{j+1,n}/(2\Delta\zeta)$$

$$S, \zeta = (S_{L+1} - S_{L-1})^{j+1,n}/(2\Delta\zeta)$$

The streamwise derivative is a one-point backward difference as seen in the development between 5.7 and 5.8:

$$F_{\xi}, \xi = (F_{\xi}^{j+1,n} - F_{\xi}^{j})_L/\Delta\xi$$

In all such differencing, the metrics are also included. In differencing the viscous terms the viscosity coefficient is also differenced. Viscosity is determined by Sutherland's Law in dimensional form:[5]

$$\mu = B(T^{3/2})/(T + S) \qquad 5.13$$

where T is dimensional temperature and

$$B = 7.3025 \times 10^{-7} \quad lbf/(ftsecR^{1/2})$$

$$S = 198.72^{0}R \quad \text{for air}$$

The block tridiagonal form of equation 5.12 is solved with a subroutine that was originally written[6] for an academic subject requiring the solution of a block tridiagonal matrix made

up of 3x3 matrices beginning at node 1. It was modified here to allow 5x5 matrices and to start at node 2.

The block tridiagonal form (5.12) in Reference 1 was solved using stored forms of the inverse matrix. The scheme of Reference 1 computes new matrices in the block tridiagonal form only at the first iteration of each streamwise step. The current code computes these matrices anew for each iteration, before solving the block tridiagonal matrix. This revised procedure does not affect the converged solution but may change the number of iterations required to obtain the converged solution.

The tridiagonal solution advances the solution in pseudo time only, from level n to level n+1. Once convergence is achieved, the solution marches from level j to j+1 by some method of prediction. For example, either the newly converged j solution becomes the first j+1 solution, or using solutions at j and j-1, extrapolation predicts the first solution at j+1. The iterations take place at one streamwise location rather than globally over the whole field as in previous iterative PNS schemes.[3,14] This, along with the separate equation of state, makes the current scheme significantly different from past algorithms.

6.0 BOUNDARY CONDITIONS

There are three distinct boundaries for the wedge problem: The wedge surface, the shock wave, and the initial data plane. Each will be covered separately.

6.1 SURFACE

Five boundary conditions are required at the wall corresponding to the five state variables. These conditions are:[1]

1) Density consistent with equation of state
2) No slip condition for u velocity: $(\rho u)_1 = 0$
3) No slip condition for w velocity: $(\rho w)_1 = 0$
4) Specified wall temperature, $T_1 = 1.0$ for Mach 3 and $T_1 = 3.0$ for Mach 15 flow.
5) Zero pressure derivative in the body normal direction, $(P,\zeta) = 0$

Here subscript 1 refers to a wall value and condition #5 is justified by a boundary layer type analysis performed at the wall.[1] The true nondimensional equation of state is $\gamma p - \rho T = 0$. However, as will be discussed in Chapter 8, a modified equation of state is taken to be:

$$\gamma p - \rho T + \theta[P,\xi + P,\zeta] = 0 \qquad 6.1$$

Because of condition #5, at the surface this reduces to

$$(\gamma p - \rho\Gamma + \theta P, \xi)_1 = 0 \qquad 6.2$$

Where θ is an "arbitrary" (small) parameter. For sufficiently small θ the model approximates the true equation of state if $(P, \xi) \approx O(1)$.

Equation 5.9 applied at the first point away from the wall (L=2) results in three matrices on the left and one vector on the right-hand side:

$$A_1 \Delta q_1 + B_2 \Delta q_2 + C_3 \Delta q_3 = G_2 \qquad 6.3$$

This vector equation represents 5 equations for the 5 state variables. Since the subscripts refer to specific nodes, $A_1 \Delta q_1$ is known from the boundary conditions. It is not part of the block tridiagonal matrix which has the form (see Equ. 5.12):

$$\begin{bmatrix} B^*_2 C_3 & & & \\ A_2 B_3 C_4 & & & \\ & \ddots & & \\ & & \ddots & \\ & & A_{LMAX-2} B_{LMAX-1} & \end{bmatrix} \times \begin{pmatrix} \Delta q_2 \\ \cdot \\ \cdot \\ \cdot \\ \Delta q_{LMAX-1} \end{pmatrix} = \begin{pmatrix} G_2 \\ \cdot \\ \cdot \\ \cdot \\ G^*_{LMAX-1} \end{pmatrix} \qquad 6.4$$

where: $B^*_2 = B_2 + f(A_1)$

$G^*_{LMAX-1} = G_{LMAX-1} - C_{LMAX}\Delta q_{LMAX}$

Since $\Delta q_1 = (q^{n+1} - q^n)_1$ is unknown (specifically q^{n+1}), this boundary condition can be treated implicitly, transformed to the delta form, and combined with the $B_2\Delta q_2$ term.

A model for condition #5 is that the pressures are equal at the wall and the adjacent node.

$$P_1 = P_2 \qquad 6.5$$

It follows therefore that

$$\Delta p_1 = \Delta p_2$$

where Δp_1 is the fifth component of Δq. The velocity terms are also straight forward since from conditions 2 and 3:

$$\Delta(\rho u)_1 = \Delta(\rho w)_1 = 0 \qquad 6.6$$

For the ρT state variable we have (from 6.2 and 6.5)

$$\Delta(\rho T)_1 = (\gamma p_1 + \theta p_1, \xi)^{n+1} - (\gamma p_1 + \theta p_1, \xi)^n$$

$$= (\gamma p_2 + \theta p_2, \xi)^{n+1} - (\gamma p_2 + \theta p_2, \xi)^n$$

$$= \Delta(\rho T)_2 \qquad 6.7$$

And finally,

$$\Delta\rho_1 = (\Delta\rho T)_1 / T_1 = (\Delta\rho T)_2 / T_1 \qquad 6.8$$

Combining equations 6.5 - 6.8 gives:

$$\Delta q_1 = \begin{pmatrix} \Delta\rho_1 \\ \Delta(\rho u)_1 \\ \Delta(\rho w)_1 \\ \Delta(\rho T)_1 \\ \Delta p_1 \end{pmatrix} = \begin{pmatrix} \Delta(\rho T)_2 / T_1 \\ 0 \\ 0 \\ \Delta(\rho T)_2 \\ \Delta p_2 \end{pmatrix} \qquad 6.9$$

The elements of the A_1 matrix which multiply these components of Δq_1 can be combined with the elements of B_2 which multiply like components of Δq_2, leading to B^*_2 of equation 6.4.

The A_1 matrix (Appendix B) is filled based on boundary conditions 1-5. The velocity components vanish, the temperature is given, and the pressure at the wall is taken from the node directly above the wall. Density is then found from the modified state equation.

$$\rho_1 = (\gamma p_1 + [\theta(P_{1,J+1,n} - P_{1,J})/\Delta\xi])/T_1 \qquad 6.10$$

6.2 Shock Boundary Conditions

There are two parts to the shock boundary condition: The location of the shock relative to the surface of the wedge, Z_{SW}, and the angle the shock surface makes with the freestream flow.

The shock angle determines the state variable values across the shock and the location of the shock is used to equalize mass flow. Since the flow is viscous, the boundary layer will force the shock surface outward and at the same time the shock angle may change as a function of streamwise location. The algorithm must allow for changes in both angle and location.

The values of the state variables at the shock must be consistent with those in the interior of the shock layer. In other words, the shock values must be coupled with the interior values. To preserve this coupled nature, the shock angle must be obtained from points interior to the shock.

One way to do this is to solve for pressure or density at the shock based on interior points and then use the Rankine-Hugoniot conditions to find the shock angle. Using backward differencing, any value at the shock can be obtained from points inward of the shock:

$$f_{LMAX} = \tfrac{1}{4}(f_{LMAX-4} - 2f_{LMAX-3} - f_{LMAX-2} + 4f_{LMAX-1}) \qquad 6.11$$

where LMAX is the node at the shock.

Use 6.11 to find $p_{s,N}$ or $\rho_{s,N}$ and the Rankine-Hugoniot conditions to find the shock angle, β. This shock angle will therefore be consistent with state variable profiles calculated at each iteration. Once $P_{s,N}$ or $\rho_{s,N}$ are found remaining variables are found from the Rankine-Hugoniot equations and the state equation. The velocities across the shock, $u_{s,N}$ and $w_{s,N}$, are found from geometric considerations detailed in Appendix C. On the first iteration, before any profiles have been calculated, either the shock pressure or density can be extrapolated from upstream values with a simple Eulerian integration. The shock values found at each iteration become the values at the n+1 level.

The global mass conservation procedure considers a problem separate from the shock value calculation and independent of the coupled nature of the shock angle. The shock location determines the maximum Z value, which determines the grid distribution, which determines the metrics. To begin this routine, the shock location must be predicted when stepping from j to j+1 prior to any iterations. At each iteration the $Z_{s,N}$ position is adjusted to equalize mass flow with freestream mass flow.

The method used to predict the shock location is based on a paper by Chausee, et al.[7] His method is written for a general three dimensional system but here reduces to:

$$Z_{sw}^{j+1} = Z_{sw}^{j} + \Delta x(\tan\beta^{j}) \qquad 6.15$$

where Δx is the streamwise step size. This prediction is used only at n=1. At further iterations, Z_{sw} adjustments are based on overall mass flow considerations.

The mass flow equalization concept was not part of Chausee's scheme. It was suggested in Reference 1 since Chausee's method results in mass flow errors of ± 2.0%.[1] Bhutta and Lewis suggest moving Z_{sw} until mass flow errors are less than ± 0.1%. Details of both the shock prediction and the mass flow calculation are presented in Appendix C.

Referring back to equation 6.4, the shock boundary concerns the $C_{LMAX}\Delta q_{LMAX}^{n+1}$ vector. And as in section 6.1, the C_{LMAX} matrix is filled using the shock values. Unlike the wall boundary condition, Δq_{LMAX}^{n+1} is known. The newly calculated shock values at each iteration become q^{n+1} and the last computed shock values are q^{n}. In this case Δq_{LMAX} can be formed explicitly:

$$\Delta q_{LMAX}^{n+1} = (q^{n+1} - q^{n})_{LMAX} \qquad 6.16$$

The right hand side of 6.4 is then corrected for this boundary condition with:

$$G^{*}_{LMAX-1} = G_{LMAX-1} - C_{LMAX}(\Delta q_{LMAX}^{n+1}) \qquad 6.17$$

6.3 INITIAL CONDITIONS

The initial data plane (IDP) is the starting point for the numerical solution. Using the current nondimensionalization, the IDP is at X=1.0. Initial values for all five state vector components are predicted or assumed from the wall to the shock. Initial data which together with the next j+1 initial prediction better satisfy the governing equations (Equ. 5.8) result in faster convergence for the first j+1 location.

In Reference 1 a blunt body starting code was used to obtain an initial condition for a blunt cone configuration; a starting solution for the present viscous wedge flow was not available. Initial conditions were based instead on inviscid flow with refinements based on the addition of a boundary layer and a mass flow calculation.

Because of this, the present initial conditions are approximations at best for a "correct" wedge flow. However, the approximations are believed to be reasonable based on an assumed boundary layer that was 10% of the initial local shock layer thickness with a specified distribution from the wall to the edge of the boundary layer. Since the mass flow must balance, the shock location and the state variable profiles were then adjusted to match the freestream mass flow with that of the IDP.

As a measure of the quality of the initial data, the right hand side of Equation 5.8 was monitored. The right hand side is simply the governing PNS system terms written at time level n. A converged solution corresponds to a vanishing right hand side to some acceptable level. The magnitude of the largest component of the right-hand side for the first iteration after the IDP indicates how well a consistent solution was achieved with the initial data and the prediction at j+1. After setting up the initial data, profiles were smoothed to blend the region at the top of the boundary layer. Figures 6.1 and 6.2 show typical initial data for Mach 3 and for Mach 15 flow. The figures are only a representation since Reynolds number was varied and therefore different initial profiles were used for different cases.

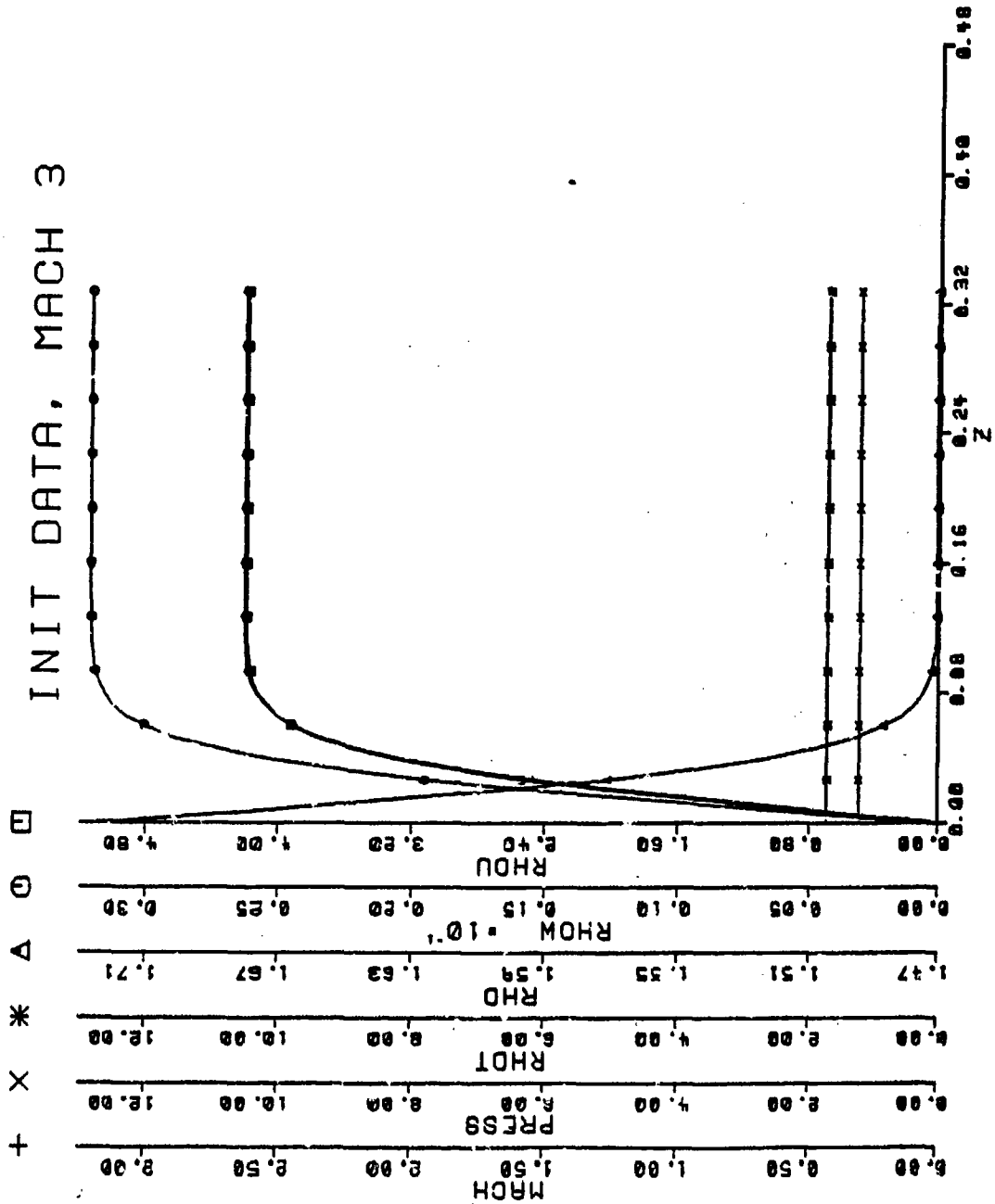


Figure 6.1 Initialization Profiles for Mach 3

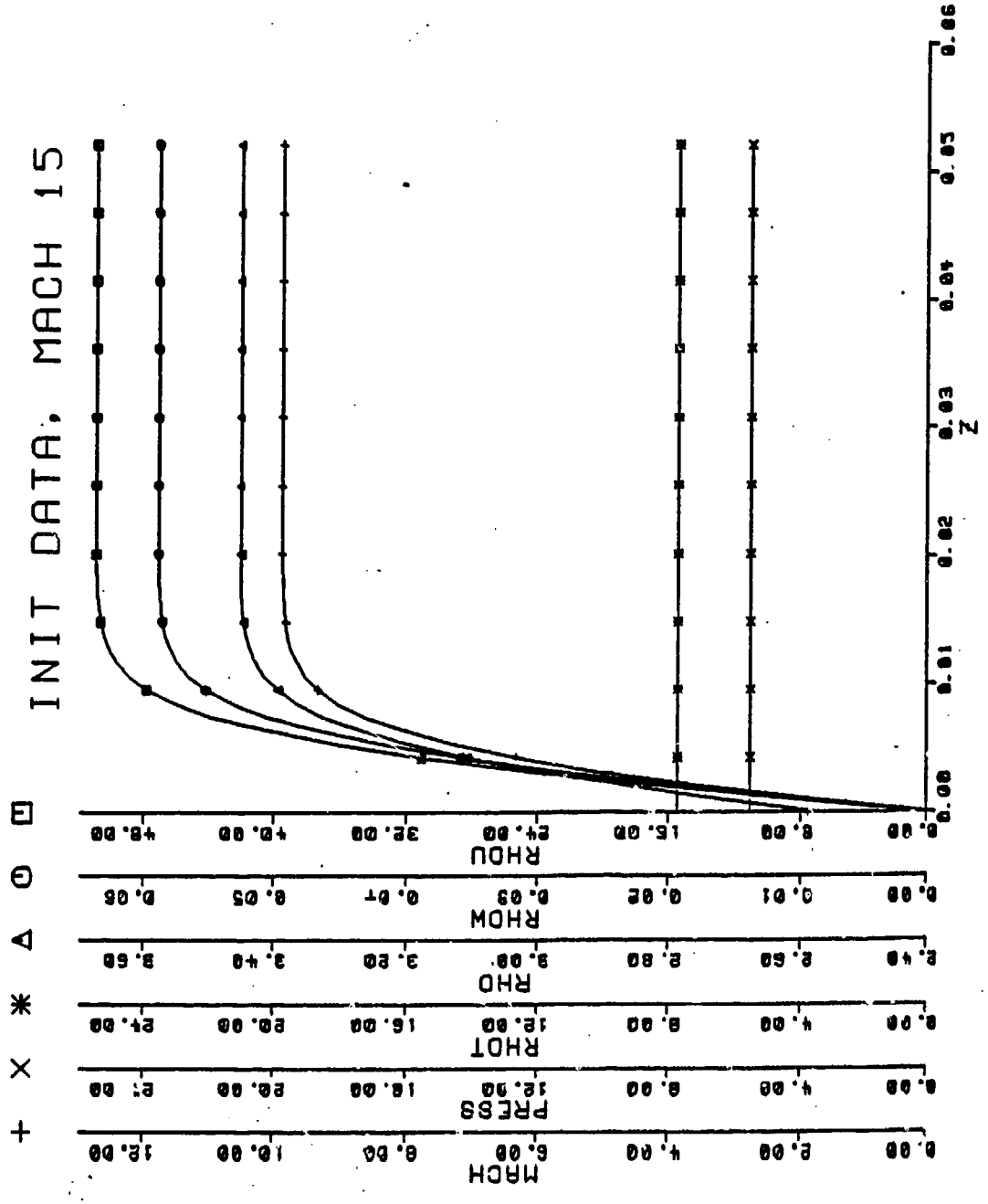


Figure 6.2 Initialization Profiles for Mach 15

7.0 SMOOTHING

The final form of the finite different equations are given by equation 5.9 with the right-hand side as in 5.8. For simplicity, this can be written as:

$$[T]^n \Delta q_L^{n+1} = [RHS]^n \qquad 7.1$$

where $[T]^n$ is the block tridiagonal matrix and $[RHS]^n$ is the right-hand side vector. The equation as it stands solves for Δq^{n+1} and as yet there is no smoothing.

As stated in Reference 1 and by Schiff and Steger,[8] central differencing produces oscillatory behavior which must be damped. Since Equation 6.1 is second order accurate in the ζ direction due to the central differencing, it is possible to add a term of $O(\Delta\zeta)^4$ as a smoothing parameter without formally affecting the second order accuracy.[1]

Reference 1 chooses the form of this smoothing parameter to be:

$$\omega[f(q^{j+1})](\Delta q)^4 \qquad 7.2$$

where f is an appropriate vector and ω is some constant.

Bhutta and Lewis[1] chose the vector f to be:

$$f = \tfrac{1}{4}[A_1 q, \xi\xi + (A_2 q, \xi\xi), \zeta - \epsilon(Mq, \xi\xi), \zeta - A_0 q, \xi\xi] \qquad 7.3$$

and $\omega = 0.$ or 1.0 for no smoothing or smoothing.

If 7.2 is added to 5.1 we obtain:

$$F_1, \xi + F_2, \zeta = \epsilon S, \zeta + H + \omega f(q^{j+1})(\Delta\xi)^2 \qquad 7.4$$

Substituting for $f(q^{j+1})$ (and using the discussion of Chapter 5) results in:

$$\begin{aligned}
&[F_1 + A_1(-\omega q, \xi\xi(\Delta\xi)^2)/4]^{j+1} \\
&+[F_2 + A_2(-\omega q, \xi\xi(\Delta\xi)^2)/4]^{j+1}, \zeta \\
&=\epsilon[S + M(-\omega q, \xi\xi(\Delta\xi)^2)/4]^{j+1}, \zeta \\
&+[H + A_0(-\omega q, \xi\xi(\Delta\xi)^2)/4]^{j+1} \\
&+F_1^j + O(\Delta\xi, (\Delta\xi)^2)
\end{aligned} \qquad 7.5$$

Now, define the quantity:

$$-\omega q, \xi\xi(\Delta\xi)^2/4 = Q^{j+1} - q^{j+1} = O(\Delta\xi)^2 \qquad 7.6$$

so that $(Q^{j+1} - q^{j+1})^2 = O(\Delta\xi)^4$ 7.7

or, to second order accuracy:

$$q^{j+1} = Q^{j+1} + \alpha Q_{,\zeta\zeta}(\Delta\zeta)^2/4$$

7.8

Consider a Taylor Series expansion of $F_1(Q)$ around q:

$$F_1(Q) = F_1(q) + F_{1,q}(Q-q) + F_{1,qq}(Q-q)^2/2 + \ldots$$
$$= F_1(q) + A_1(-\alpha q_{,\zeta\zeta}(\Delta\zeta)^2)$$
$$+ O(\Delta\zeta)^4$$

Similar constructs can be obtained for F_2, S and H and in terms of an intermediate solution Q and Equation 7.5 can be written as:

$$F_1(Q)^{j+1} + F_2(Q)^{j+1}{}_{,\zeta} = \epsilon[S(Q)^{j+1}]_{,\zeta} + H(Q^{j+1}) + F_1{}^j$$

7.9

The intermediate solution, Q, is related to q by Equation 7.8. Comparing Equation 7.9 to the development in Chapter 5, we can see that 7.9 in terms of Q can be handled like 5.8 in terms of q. In other words, the governing equation can now be written as

$$[A\Delta Q^{n+1} + B\Delta Q^{n+1} + C\Delta Q^{n+1}]^{j+1} = G_L{}^{j+1,n}$$

7.10

where A, B, C are the tridiagonal matrices, or

$$[T]^n \Delta Q^{n+1} = [RHS]^n \qquad 7.11$$

This is the equation to be coded and solved. Once $\Delta Q^{j+1,n+1}$ is calculated, $Q^{j+1,n+1}$ would be found from:

$$Q^{j+1,n+1} = Q^{j+1,n} + \Delta Q^{j+1,n+1} \qquad 7.12$$

and once $Q^{j+1,n+1}$ is found, the real solution $q^{j+1,n+1}$ would be found from 7.8 as:

$$q_L{}^{j+1,n+1} = [\omega(Q_{L+1} + Q_{L-1})/4 + (1-\omega/2)Q_L]^{j+1,n+1} \qquad 7.13$$

When solving the block tridiagonal form of 7.10 or 7.11, the solution is in terms of the vectors ΔQ^{n+1}. The solution at level n, which is used to calculate the right hand side of 7.10, is in terms of the true solution, q. The only term that is known is ΔQ^{n+1}, the solution of the tridiagonal matrix.

ΔQ^{n+1} must be changed to Δq^{n+1} before it can be added to q^n. In other words, ΔQ^{n+1} is smoothed to Δq^{n+1} which is then added to q^n to obtain the upgraded solution

$$q^{n+1} = q^n + \Delta q^{n+1} \qquad 7.14$$

This is very similar to 7.12 but with Q in place of q. If equation 7.13 at level n+1 is subtracted from the same equation at level n, the following is obtained.

$$\Delta q_L{}^{j+1,n+1} = \omega(\Delta Q_{L+1} + \Delta Q_{L-1})^{j+1,n+1}/4 + (1-\omega/2)\Delta Q_L{}^{j+1,n+1} \qquad 7.15$$

Δq^{n+1} is now in terms of $\Delta \dot{Q}^{n+1}$
which is the actual result of the tridiagonal solution for use in Equation 7.14.

Equation 7.15 is used from nodes L = 2 to LMAX-1. Care is needed at nodes 2 and LMAX-1. At node LMAX-1, ΔQ_{L+1} is ΔQ at the shock. At node 2, ΔQ_{L-1} is ΔQ at the wall and both must be accounted for.

At node 2, as was shown in Chapter 6:

$$\Delta Q_{L-1} = \begin{pmatrix} \Delta\rho_1 \\ \Delta(\rho u)_1 \\ \Delta(\rho w)_1 \\ \Delta(\rho T)_1 \\ \Delta p_1 \end{pmatrix} = \begin{pmatrix} \Delta(\rho T)_2 / T_1 \\ 0 \\ 0 \\ \Delta(\rho T)_2 \\ \Delta p_2 \end{pmatrix}$$

At node 2 then, Equation 7.15 becomes: (For components 4, 5)

$$\Delta q_L^{j+1,n+1} = [\omega\Delta Q_{L+1}/4 + (1-\omega/4)\Delta Q_L]^{j+1,n+1} \qquad 7.16$$

For component 1, ΔQ_{L-1} is the fourth component of ΔQ_2, divided by wall temperature. For components 2 and 3, 7.15 remains the same.

At node LMAX-1, as was also presented in Chapter 6:

$$\Delta Q_{L+1}^{n+1} = \Delta Q_{LMAX}^{n+1} = Q_{LMAX}^{n+1} - Q_{LMAX}^{n} \qquad 7.17$$

This is known at the shock and substituted into 7.15 during smoothing.

The calculated ΔQ^{n+1} in the tridiagonal algorithm becomes the basis of convergence. If all ΔQ^{n+1} approach zero, the solution at the current j+1 node is converging properly. The criteria used here was that all ΔQ^{n+1} must be less than or equal to 0.0001 before moving on to the next streamwise node.

8.0 STABILITY AND EIGENVALUE ANALYSIS

Since this method is implicit, it is expected that for reasonable step sizes:

$$\Delta x \text{ or } \Delta \xi$$

the finite difference algorithm will be unconditionally stable. In Reference 1, the step size variations were said to be related to changes in grids, shock propagation accuracy and solution convergence rates[1]. No detailed algorithm was given. However, the stability considerations to be discussed here are not related to the discrete differencing but to the actual physical nature of the problem, i.e., has the current formulation eliminated the elliptic (subsonic) region of the domain?

Recall Equation 5.7:

$$\begin{aligned}
&[F_i^{j+1,n} + A_i^{j+1,n}\Delta q^{n+1}],\xi \\
&\quad + [F_e^{j+1,n} + A_e^{j+1,n}\Delta q^{n+1}],\zeta \\
&\qquad = \epsilon[S^{j+1,n} + M^{j+1,n}\Delta q^{n+1}],\zeta \\
&\qquad\quad + [H^{j+1,n} + A_e^{j+1,n}\Delta q^{n+1}]
\end{aligned} \tag{8.1}$$

Where A_i, A_e, M, and A_e can theoretically change in both the ξ and ζ directions. Now assume, as in Reference 1, that these coefficients are frozen. This is reasonable if changes from n to n+1 are not large. The above equation is now:

$$a_1 \Delta q, \xi + a_2 \Delta q, \zeta$$

$$-\epsilon m \Delta q, \zeta - a_3 \Delta q = G \qquad 8.2$$

where a_1, a_2, m, and a_3 are frozen.

As a further simplification,[1] examine the viscid and inviscid limits separately. Equation 8.2 is too difficult to analyze as is. The inviscid and viscous limits are simpler and may be analyzed separately.

8.1 INVISCID LIMIT

The inviscid limit of 8.2 can be written as

$$a_1 \Delta q, \xi + a_2 \Delta q, \zeta + K(\Delta q, \xi, \zeta) = 0$$

The stability analysis of the system of equations now requires that the inverse of a_1 be formed and multiplied throughout giving:

$$\Delta q, \xi + a_1^{-1} a_2 \Delta q, \zeta + a_1^{-1} K = 0 \qquad 8.3$$

The eigenvalues of the $a_1^{-1} a_2$ matrix now determine the marching stability of equation 8.3.[1] Even though 8.3 is a much simplified version of 5.9, it is a form suitable to mathematical analysis. If 8.3 is stable then the full equation may also be stable; if 8.3 is unstable, the full equation certainly will be unstable.

As a further simplification, assume a rectangular grid for this analysis so that the metrics become

$$\xi_x = 1 \qquad \xi_z = 0$$

$$\zeta_x = 0 \qquad \zeta_z = 1$$

$$J = 1$$

The state vector is unchanged.

Introduce the following differential equation of state:[4]

$$\gamma p - \rho T + \theta(P_\xi + P_\zeta) = 0 \qquad 8.4$$

This becomes a generalized fifth equation in the system as was discussed in Chapter 3. Without 8.4 the F_1 and F_2 vectors become

$$F_1 = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho w u + p \\ (T/(\gamma-1)+V^2/2)\rho u \\ 0 \end{pmatrix}$$

$$F_2 = \begin{pmatrix} \rho w \\ \rho uw + p \\ \rho w^2 + p \\ (T/(\gamma-1)+V^2/2)\rho w \\ 0 \end{pmatrix}$$

The matrices a_1 and a_2 are formed from the partial derivatives of the flux vectors with respect to the state vector (Appendix B).

An eigenvalue determination requires the inverse of a_1 and multiplication by that inverse. However, the last row of a_1 consists of zeros since the last component of F_1 is zero so the inverse of a_1 is undefined. To circumvent this, Bhutta and Lewis invented Equation 8.4.

Using 8.4, the fifth component of F_1 and F_2 becomes θp (Equations 3.16) and the last row of a_1 and a_2 becomes

$$0 \quad 0 \quad 0 \quad 0 \quad \theta$$

Now the inverse of a_1 can be formed and the eigenvalues of $a_1^{-1}a_2$ can be found. Although Reference 1 indicates that 8.4 was not used in the actual computer code, a subsequent paper by Bhutta and Lewis[11] does show solutions for different values of θ and seems to indicate equation 8.4 was included. This explains the inclusion of this equation of state. The intent was to

investigate different values of θ and its possible effect on the solution.

The eigenvalues of $a_1^{-1}a_2$ (based on $\theta>0$ but for the limit θ approaching 0) are:

$$1, \quad w/u, \quad w/u, \quad w/u, \quad w/u \qquad 8.5$$

Reference 1 incorrectly indicates zero in place of unity. The eigenvalues of the 4 x 4 matrices, i.e. without the equation of state as the fifth equation of the formal system, are:[4]

$$w/u, \quad w/u, \quad uw \pm a(u^2+w^2-a^2)^{1/2}/(u^2-a^2) \qquad 8.6$$

The essential difference, of course, is that in 8.5 the eigenvalues are always real so that hyperbolic/marching behavior is indicated. In subsonic regions, the second set of eigenvalues (Eqn. 8.6) become imaginary and elliptic behavior is indicated.

Assuming for the moment the analysis of Reference 1 is correct, the new scheme appears to be stable for marching. In contrast to previous PNS schemes which make pressure assumptions in the subsonic sublayer to eliminate the instability, the new method requires none. In other words, even though the same physical problem is solved by both methods, mathematical reformulation seems to grant stability to the new scheme without

any assumptions in the θ=0.0 limit, for which the 8.6 eigenvalues are to be expected.

8.2 VISCOUS LIMIT

In the viscous limit, equation 8.2 becomes:

$$a_1 \Delta q_{,\xi} - \epsilon m \Delta q_{,\zeta} + K = 0 \qquad 8.7$$

Using the same θ rationale as in the inviscid limit, the viscous eigenvalues are[1]:

$$0, \quad 0, \quad 0, \quad 3\epsilon\mu/(Pr\rho u), \quad 10\epsilon\mu/(3\rho u) \qquad 8.8$$

In previous PNS schemes the viscous eigenvalues are[4]

$$0, \quad 4\mu/(3\rho u), \quad \sigma_3, \quad \sigma_4$$

Where

$$\sigma_{3,4} = \mu/(2\rho(u^2-a^2))((u+D/(uE)) \pm [(u+D/(uE))^2 - 4D/E]^{1/2})$$

$$D=\gamma u^2 - a^2 \qquad E=Pr\mu/k$$

and: k = coefficient of thermal conductivity.

The discussion concerning marching stability is similar to that for the inviscid limit. The only difference is that

positive viscous eigenvalues require positive u-velocity components. This says that reversed flow is unstable, in agreement with accepted practice.

8.3 DISCUSSION AND ANALYSIS

To this point the results and eigenvalues may seem surprising given that mathematical reformulation seems to remove the marching instability when $\theta=0$. The analysis was first presented in Reference 1 and the results of equation 8.5 check mathematically. Whether or not such an analysis was justified will be examined here using a small perturbation and linear stability analysis.

Stability for the mixed differential/algebraic system of Equation 8.3 is unclear since $\theta=0$ implies that the necessary matrix operations for eigenvalues are undefined. When θ is included to form a totally differential system, the θ approaching 0 limit does not yield the correct eigenvalues (Eqn 8.6). There is a paradox involving the parameter θ, presumably related to the singular perturbation form of the modified state equation. This reasoning leads to the following analysis.

8.3.1 Small Perturbation Analysis*

Consider a simple model problem which appeared in Reference 11:

$$\begin{aligned} u_x - v_y &= 0 \\ v_x - u_y + 2v_y + w_y &= 0 \\ \epsilon w_x &= a^2 u - v \end{aligned} \qquad 8.9$$

Equation 8.9 loosely approximates and is representative of the governing PNS equations with a modified ($\epsilon > 0$) equation of state.

If $\epsilon = 0$ 8.9 is equivalent to

$$\begin{aligned} u_x - v_y &= 0 \\ v_x + (a^2-1)u_y + 2v_y &= 0 \end{aligned} \qquad 8.10$$

which can be written as

$$q_x + Aq_y = 0$$

where $q = [u, v]^T$ and

$$A = \begin{bmatrix} 0 & -1 \\ (a^2-1) & 2 \end{bmatrix}$$

---

* Suggested by Dr. Judson R. Baron and Dr. Saul S. Abarbanel

The eigenvalues of A are:

$$\lambda_{1,2} = 1 \pm (2-a^2)^{1/2} \qquad 8.11$$

and are imaginary if $a > 2$.

The eigenvectors are:

$$e_1 = [1, -\lambda_1]^T \quad , \quad e_2 = [1, -\lambda_2]^T$$

If ϵ is not zero, the eigenvalues of 8.9 become

$$\lambda_{1,2,3} = 0, \; 1 \pm (2)^{1/2} \qquad 8.12$$

and the eigenvectors are:

$$e_1 = [1, 0, 1]^T \quad , \quad e_2 = [1, -\lambda_2, 0]^T \quad , \quad e_3 = [1, -\lambda_3, 0]^T$$

The eigenvalues are real and have no "time like" constraint. Similarly, the eigenvalues of Equation 8.5 were unconditionally real after Bhutta and Lewis used their θ not equal to zero assumption to allow matrix inversion and took the limit as θ approached 0.0.

The preceding discussion confirms the Bhutta and Lewis algebra leading to 8.5. The essential point is that the eigenvalues appear to be independent of ϵ or θ even though ϵ or θ cannot apriori be set equal to zero to find the eigenvalues. What are the proper eigenvalues as ϵ or θ approach zero in view of the differing results of 8.11 and 8.12 (or 8.5 and 8.6)? Is it not necessarily correct on physical grounds to assume the eigenvalues when $\epsilon = 0$ and ϵ approaches zero must be identical?

Consider the implications of assuming a regular perturbation based on ϵ. First, differentiate the second equation of 8.9 w.r.t. x and the third w.r.t y. Eliminate w_{xy} to get:

$$u_x - v_y = 0$$
$$v_{xx} - u_{yx} + 2v_{yy} + (a^2 u_y - w_y)/\epsilon = 0$$

Then with w_y from the second equation and u_{yx} from the first equation of 8.9:

$$u_x - v_y = 0$$
$$\epsilon[v_{xx} - v_{yy} + 2v_{yx}] + a^2 u_y + v_x - u_y + 2v_y = 0 \qquad 8.13$$

Assume a regular perturbation, say:

$$u = u_0 + \epsilon u_1 + \epsilon^2 u_2 + \ldots$$
$$v = v_0 + \epsilon v_1 + \epsilon^2 v_2 + \ldots \qquad 8.14$$

and examine the eigenvalue basis for different orders of ϵ.

Substituting 8.14 into 8.13 and collecting terms according to like powers of ϵ gives:

$O(\epsilon^0)$ $\quad u_{0x} - v_{0y} = 0 \qquad 8.15a$
$$v_{0x} - u_{0y} + 2v_{0y} + a^2 u_{0y} = 0$$

$O(\epsilon^1)$ $\quad u_{1x} - v_{1y} = 0$

$$v_{1x} - u_{1y} + 2v_{1y} + a^2 u_{1y} = -(v_{0xx} - v_{0yy} + 2v_{0xy}) \qquad 8.15b$$

$O(\epsilon^2)$ $\quad u_{2x} - v_{2y} = 0$

$$v_{2x} - u_{2y} + 2v_{2y} + a^2 u_{2y} = -(v_{1xx} - v_{1yy} + 2v_{1xy}) \qquad 8.15c$$

and so on.

Thus, in general, for $O(\epsilon^n)$, each set of equations can be written as:

$$L(q_N) = R(v_{N-1})$$

where the L operator is identical for the equations of all orders.

The point is that if the introduction of ϵ allows a regular perturbation as in 8.14, the eigenvalues should be identical for all orders of ϵ, including the zeroeth order. The Bhutta and Lewis eigenvalues are not identical for either the model ($\epsilon = 0, \ll 1$) problem 8.9 or the actual ($\theta = 0, \ll 1$) problem 5.9.

It must be concluded that the Bhutta and Lewis eigenvalues are inconsistent. The normal subsonic sublayer instability remains in the PNS equations. Solutions constructed without any modifying assumptions to this sublayer should break down due to this instability.

8.3.2 Linear Stability Analysis*

Bhutta and Lewis erred in their inviscid stability analysis since in Equation 8.3 they did not consider the term $K(\Delta q, \xi, \zeta)$, which is also a function of the state vector, must also be included in the stability analysis.

Rewrite equation 8.3 as

$$Aq_x + Bq_y = Cq$$

where $q = [\rho, u, w, T, p]^T$. Using a linear Fourier stability technique, set

$$q = qe^{i(kx + ly)}$$

Then

$$\det(kA + lB - iC) = 0$$

Here the A and B matrices are the partial derivatives of the X and Z flux vectors with respect to the state vector above (Appendix B). The C matrix is the partial derivative of the state equation vector, H, with respect to q above.

---

*Suggested by Dr. Michael B. Giles

On evaluating the determinant it is found that

either

$$k = (i/\theta)(u^2-a^2)/u^2$$

which is unstable if M<1

or

$$k = [uwl \pm ((uvl)^2 - (a^2 - u^2)^2 l^2)^{1/2}]/(a^2-u^2)$$

$$l = [uw \pm a(u^2 + v^2 - a^2)]/(a^2-u^2)$$

which are unstable if M<1

The conclusion from both this and the previous section 8.3.1 is that the Bhutta and Lewis method does retain the weak elliptic region in the subsonic sublayer. Unless appropriate measures are taken, their method should break down at some downstream location.

9.0 RESULTS

Based on the conclusions of Chapter 8, the present scheme should prove to be unstable at some point downstream. However, the iterative and added equation aspects of the scheme make it so much different than previous PNS formulations[6,9,13,14,15] that the instability may manifest itself in different ways.

In order to test the performance of the present code over the wedge geometry, cases were run at Mach numbers 3 and 15 and Reynolds numbers ranging from approximately 4×10^3 to 1×10^7 at the IDP. Freestream data corresponding to 80,000 feet altitude was used when dimensional temperature was required in the Sutherland law, and to define a typical reference length from Reynolds number. The Mach number, Reynolds number and Prandtl number completely define the air flow, irrespective of altitude and are the only relevant inputs. A Prandtl number of 0.72 was used.

The results of the numerical tests are summarized in Tables 9.1, 9.2, and 9.3. These tables represent a progression in grid clustering. The results in Table 9.1 were carried out on an evenly spaced 50 point grid with the first node point 2.0% of the shock layer thickness from the wall. Table 9.2 used a clustered 50 point grid with the first node point 0.6% from the wall. Table 9.3 used a clustered 100 point grid with the first node point 0.3% from the wall. The clustering parameter, s, cf

Chapter 4, was 1.0×10^{-3} for Table 9.1 and 3.0 for Tables 9.2 and 9.3.

The progression from smooth to clustered grids was selected so as to obtain general performance characteristics of the scheme. Certain cases were chosen from Table 9.1 and rerun on tighter grids to obtain the results of Table 9.2. In this manner, similar cases were comparable with respect to their performance covering general characteristics over the entire shock layer to very specific detail within the boundary layer and in particular within the subsonic sublayer.

TABLE 9.1 SMOOTH GRID RESULTS

ALL VARIABLES RECORDED AT INITIAL DATA PLANE

CASE	MACH NO.	RE_L	Z_{s_L}/Z_{s_N}	Δ_y	δ_s	#STEPS	FINAL X
1	3	1.6×10^{8}	0.04	0.03	0.013	>150	5.5
2	3	2.3×10^{4}	0.10	0.03	0.021	18	1.54
3	3	4.3×10^{3}	0.21	0.03	0.045	10	1.30
4	3	9.7×10^{4}	0.05	0.03	0.017	70	3.21
5	15	3.8×10^{6}	0.05	0.03	0.0003	>100	4.00
6	3	1.6×10^{3}	0.04	0.001	0.013	17	1.017
7	3	1.6×10^{5}	0.04	0.001	0.013	47, θ=0.01	1.047
8	3	1.6×10^{5}	0.04	0.001	0.013	69, θ=0.10	1.069
9	3	1.6×10^{5}	0.04	0.06	0.013	82	5.92
10	15	6.0×10^{6}	0.04	0.06	0.0003	>100	7.00
11	3	1.6×10^{5}	0.04	0.10	0.013	50	6.00
12	3	1.6×10^{5}	0.02	0.10	0.013	51	6.10
13	15	1.1×10^{7}	0.03	0.03	0.0002	>160	5.80
14	15	8.9×10^{6}	0.10	0.03	0.0006	15	1.45
15	15	1.1×10^{5}	0.25	0.03	0.00128	3	1.09

TABLE 9.2 50 POINT CLUSTERED GRID RESULTS

16	3	1.6×10^{5}	0.04	0.03	0.006	10	2.30
17	3	2.3×10^{4}	0.10	0.03	0.019	8	1.24
18	3	4.3×10^{3}	0.21	0.03	0.05	1	1.03
19	3	4.8×10^{4}	0.07	0.01	0.0125	30	1.30
20	3	1.5×10^{5}	0.04	0.01	0.006	30	1.30
21	3	2.5×10^{6}	0.01	0.01	0.004	>100	2.00
22	15	6.0×10^{6}	0.04	0.001	0.00017	12	1.012
23	15	6.0×10^{6}	0.04	0.0001	0.00017	>55	1.0055

TABLE 9.3 100 POINT CLUSTERED GRID RESULT

24	15	6.0×10^{6}	0.04	0.0001	0.00015	55	1.0055

As the grids become increasingly clustered near the wall, the apparent height of the subsonic layer changes slightly for equivalent boundary layer thicknesses. This is due, of course, to corresponding node location changes near the wall as the number of nodes increases. Cases 1, 5, 10, 13, 21, and 23 all could have continued for a larger number of steps than indicated. That does not mean that an eventual break down is avoidable. For example, case groups (1, 2 and 3) and (13, 14 and 15) show the resulting behavior for increasing boundary layer thickness with a constant step size. Cases 1 and 13 were concluded at a sufficiently large X such that the information gained showed a trend.

Figure 9.1 shows a plot of the number of marching steps before breaking down versus the height of the initial subsonic layer for Mach = 3 and a stepsize of 0.03. Figure 9.2 is a similar plot for Mach number 15 and stepsize of 0.03. The same results are evident in cases 16 through 21. A dramatic rise occurs in the number of possible steps at constant stepsize as the subsonic layer decreases. The exponential like behavior as the subsonic layer shrinks suggests an unlimited number of steps would be achievable for an inviscid flow.

Cases 9 and 10 provide related information. All other things being equal, the solution can be marched farther downstream at high Mach numbers because the subsonic layer is smaller. This is apparent in cases 4 and 5 as well.

It is interesting to compare the solutions for Mach 3 and Mach 15 in cases 1 and 13. Figures 9.3 and 9.4 show the profiles of state variables and Mach number across the shock layer at 150 steps for Mach 3 and 15 respectively. Notice how sharp the transition zone between the boundary layer and the inviscid external layer has become when compared to the initial data (Figs. 6.1 and 6.2).

Figures 9.5 and 9.6 show the right hand side of the governing equations (Eq. 5.8) versus vertical node. In both cases the energy equation shows the largest departure in magnitude from zero, followed by the X-momentum equation. Recall that the right hand side should approach zero in the limit of convergence at each step. A nonvanishing right hand value at any node indicates the extent to which the governing equations are not being satisfied at that node. Typically it was found that the right hand side was of O(1) for the first iteration of the first streamwise step after the IDP and became smaller proceeding in the stramwise direction until the scheme broke down. The largest values remain within the boundary layer and are caused primarily by an inconsistent set of initial conditions.

Figures 9.7 and 9.8 show the shock surface and wall pressure for the two cases. The inviscid shock has been added for reference. The large jump in wall pressure could be attributable to errors in the initial pressure. Finally, Figures

9.9 and 9.10 are enlargements of the boundary layer regions for the two cases. In both cases the reversed density gradient should be noted and in Figure 9.9 note that the ρu gradient is lessening and eventually reversed flow must be anticipated.

In fact, it was found that independent of the break down location, the sheme usually broke down in a similar way. Figures 9.11 through 9.17 trace the development of case 4 for selected downstream locations (X = 1.12, 1.30, 1.90, 2.80, 3.10). Notice particularly the behavior of ρu and ρ.

At step 30, Figure 9.13, the density has a sharp gradient and the ρu variable has less of a gradient near the wall.

Steps 60 and 70 show the u-velocity approaching separated flow. Figures 9.15 and 9.17 show the right hand side values (Equ. 5.8) for steps 60 and 70 after they have converged. Notice that the magnitudes in Figure 9.15 show that the governing equations are being satisfied while those in Figure 9.17 indicate that the solution is beginning to break down. In the same manner all cases but one which become unstable do so by converging to separated flow. Figure 9.17a details the boundary layer at the onset of reversed flow for Case 20. The large density spike is characteristic of the solution approaching separation. The governing equations remain satisfied until the break down is imminent.

Figure 9.18 shows the distribution of pressure across the shock layer for case 9, which also converges to separated flow. The profiles are plotted at steps 2, 4, 8, 16, 32 and 64 down the wedge (X=1.12, 1.24, 1.48, 1.96, 2.92, 4.84). Figure 9.18a is a blowup of 9.18. The profile at step 64 (X = 4.84) is approaching the pressure profile of Figure 9.3 which had gone 150 steps with a slightly higher Reynolds number. This seems to show that for nearly identical initial conditions, the scheme predicts consistent solutions independent of the two different streamwise stepsizes used. (Δx=0.03 in Figure 9.3 and Δx=0.06 in Figure 9.18).

Another interesting phenomenon is observed with an unstable Δx. Consider case 6. This is the only case that did not break down due to flow separation. Figures 9.19 and 9.19a show that ρu is not tending toward a decreasing gradient. It is also interesting to note that the errors are at or below the sonic line, which is shown for reference. A possible reason for this is suggested in Chapter 10. Figure 9.20 shows the growing magnitudes of the right hand side of the governing equations (Eq. 5.8) for case 6.

Finally, consider case 24 of Table 9.3. Recall that this result was obtained on the most clustered, 100 point, grid. Previous PNS schemes[9,13,15] have documented the phenomenon of departure solutions. Departure behavior is found in PNS codes that do not include a pressure assumption and is characterized by

an exponential drop in wall pressure associated with the subsonic ellipticity. Figure 9.21 shows the wall pressure distribution for case 24 and Figure 9.22 shows a u-velocity component profile for two downstream X locations. Both figures are consistent with previously demonstrated departure behavior[9,13,15] including the rapid development of the pressure drop. Figure 9.23 shows the right hand side distribution from equation 5.8 and indicates that the largest errors appear in the energy equation. This again is consistent with departure behavior[9,13,15]. Figure 9.24 displays the lower portion of the shock layer at departure. Note the large negative gradient in ρw. Compare this with Figures 9.25 and 9.26, which are from case 23. The only difference between the two cases is the grid clustering. They are displayed at the same downstream location, X=1.0055. Despite this, case 24 indicates departure while case 23 does not. Note particularly that the negative ρw gradient exists in Figure 9.25. But the pressure and energy behave quite differently than those in Figure 9.24. This is consistent with and appears to validate the claim of Reference 15 which states that mesh spacing and stepsize must be of $O(Re^{-3/8})$ or less in order to pick up departure behavior.

The indications from cases 1-5 and 9-23 is that even when classic departure behavior is not evident, the instability manifests itself by converging to a separated flow. Figure 9.27 shows the right hand side distribution from equation 5.8 of case 23. Of importance is that even though departure behavior is

suppressed errors do grow in the energy equation, which is consistent with case 24.

Figure 9.28 provides a last comparison. Pressure distribution profiles in the lower part of the shock layer are shown at different X locations during departure. Notice that as the solution marches downstream a pressure disturbance moves outward away from the surface in the shock layer. This may be compared with the behavior in Figure 9.17, which was run at Mach 3 and also exhibits an outward moving pressure wave. Figure 9.17 represents a somewhat more stable solution since as the solution marches downstream, the pressure tends towards constancy and is consistent with known wedge flow. Pressure in Figure 9.28, on the other hand, moves away from constancy but the underlying instability in both cases possibly induces similar behavior.

The cases shown have been selected because they provide a representative characterization of the behavior of the current scheme. They appear to imply that the scheme is unstable, that it is unstable due to departure behavior, and even when classic departure behavior is not evident due to grid sizing or step size, the underlying instability causes convergence to separated flow.

The results are not exhaustive but they do serve to outline the performance of the current algorithm. Because of the iterative nature of this particular algorithm (Chap. 5) the

classical PNS pressure modeling could not be included. Several of the previous sublayer assumptions were mentioned in Chapter 1 and it would have been informative to use similar sublayer assumptions in the current scheme, hoping to achieve a stable algorithm.

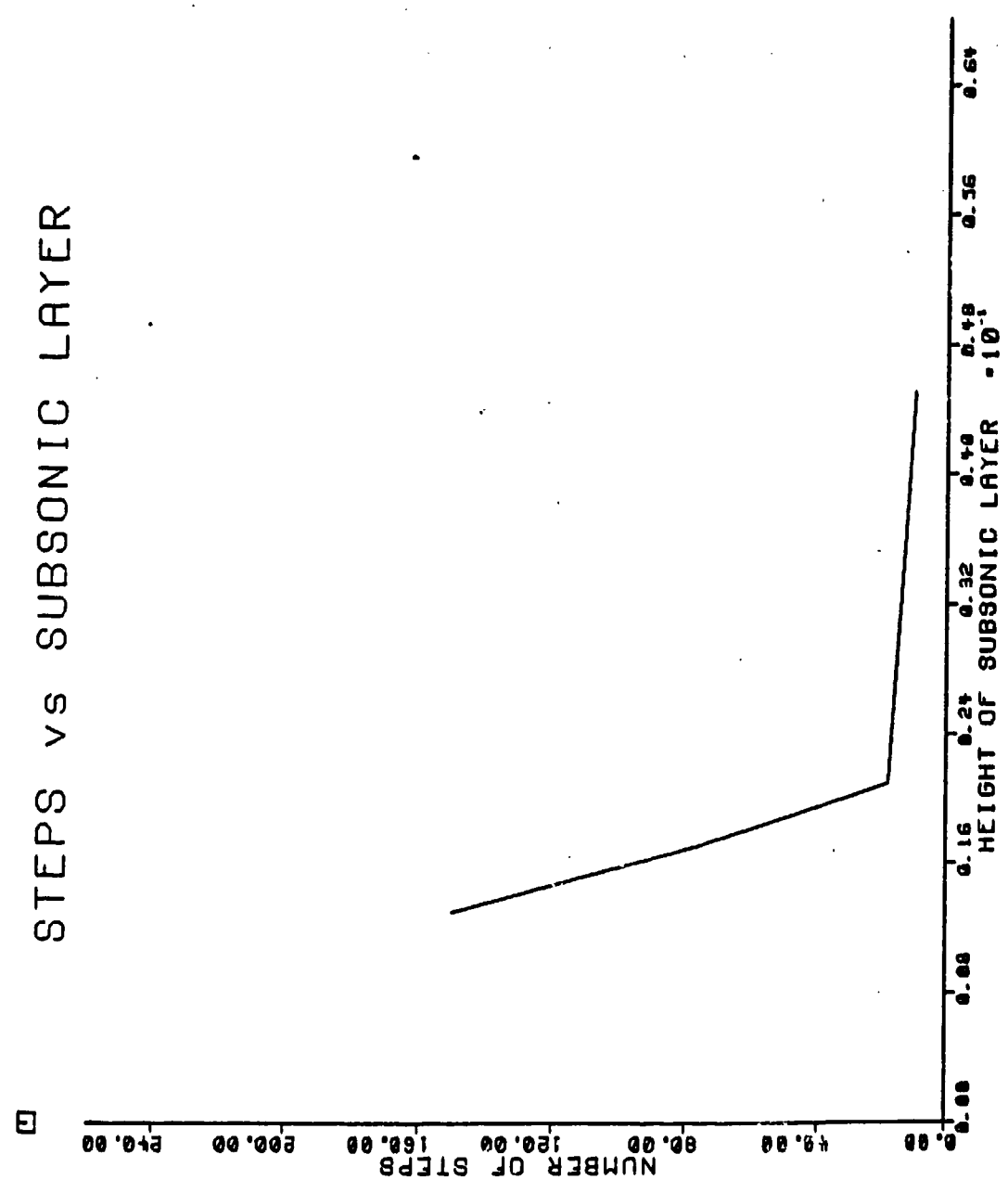


Figure 9.1 Streamwise Steps vs Subsonic Layer Height at Mach 3, Δx=0.03

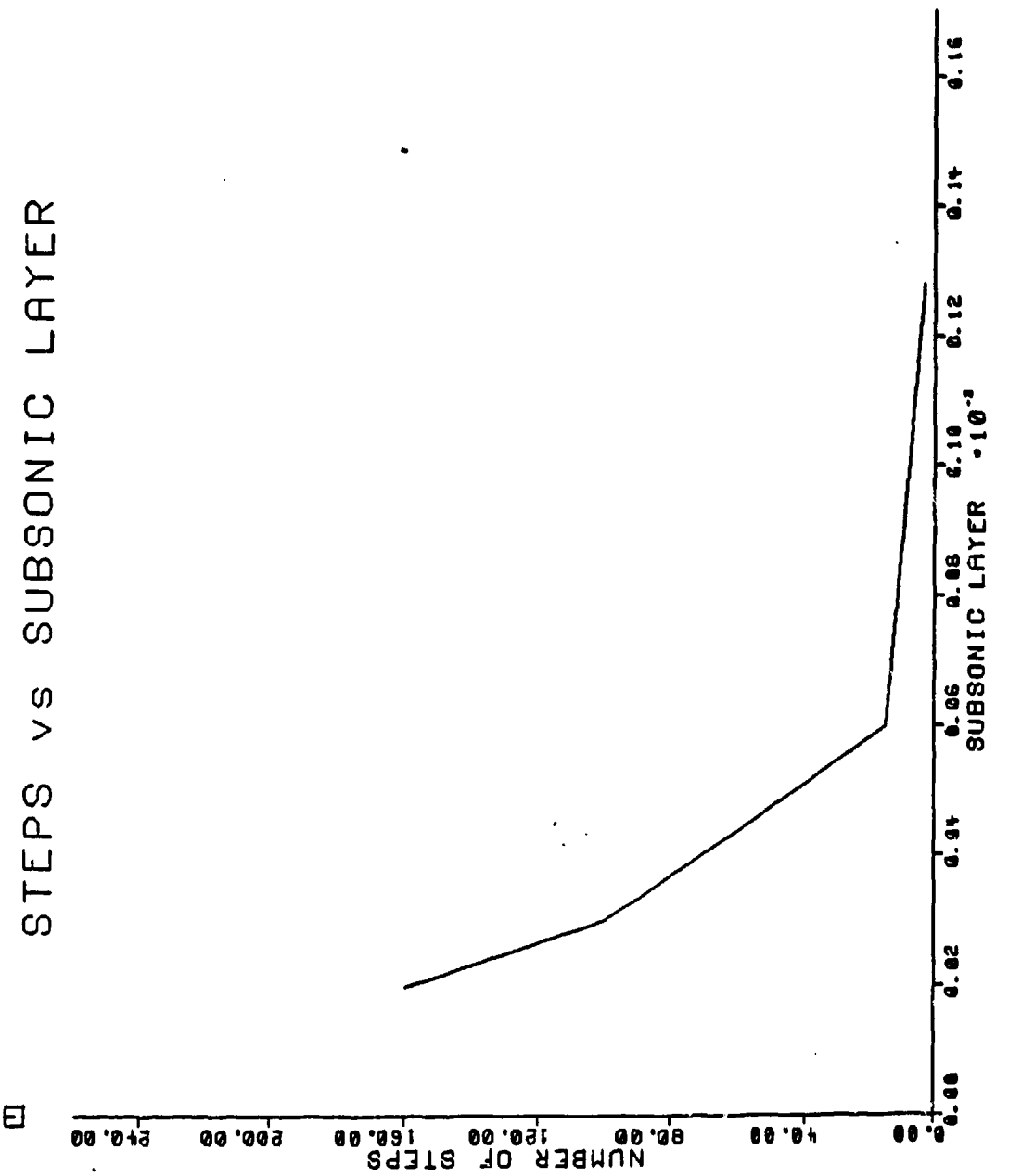


Figure 9.2 Streamwise Steps vs Subsonic Layer Height at Mach 15, $\Delta x = 0.03$

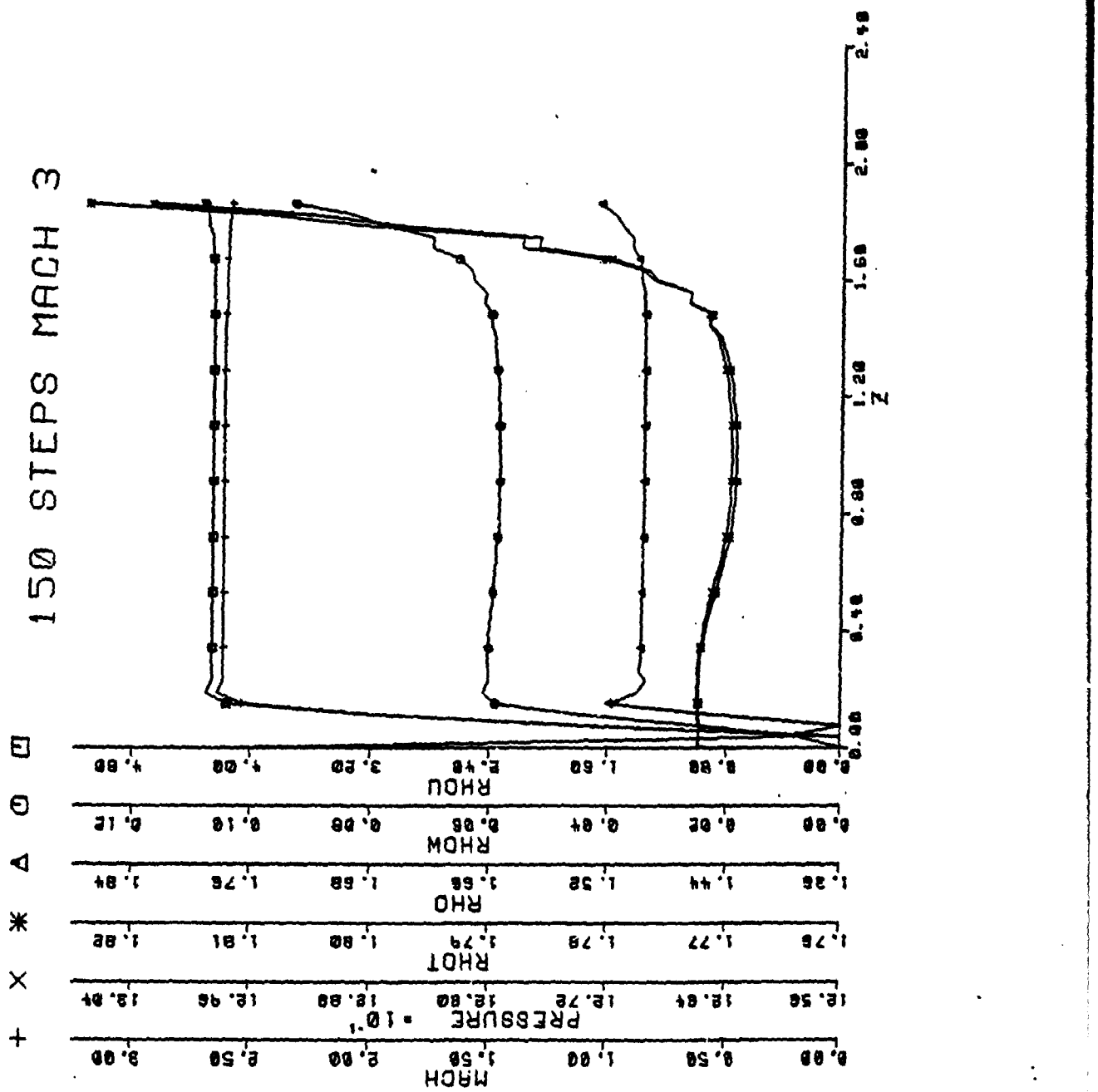


Figure 9.3 State Variable Profiles after 150 Steps at Mach 3, $\Delta x = 0.03$

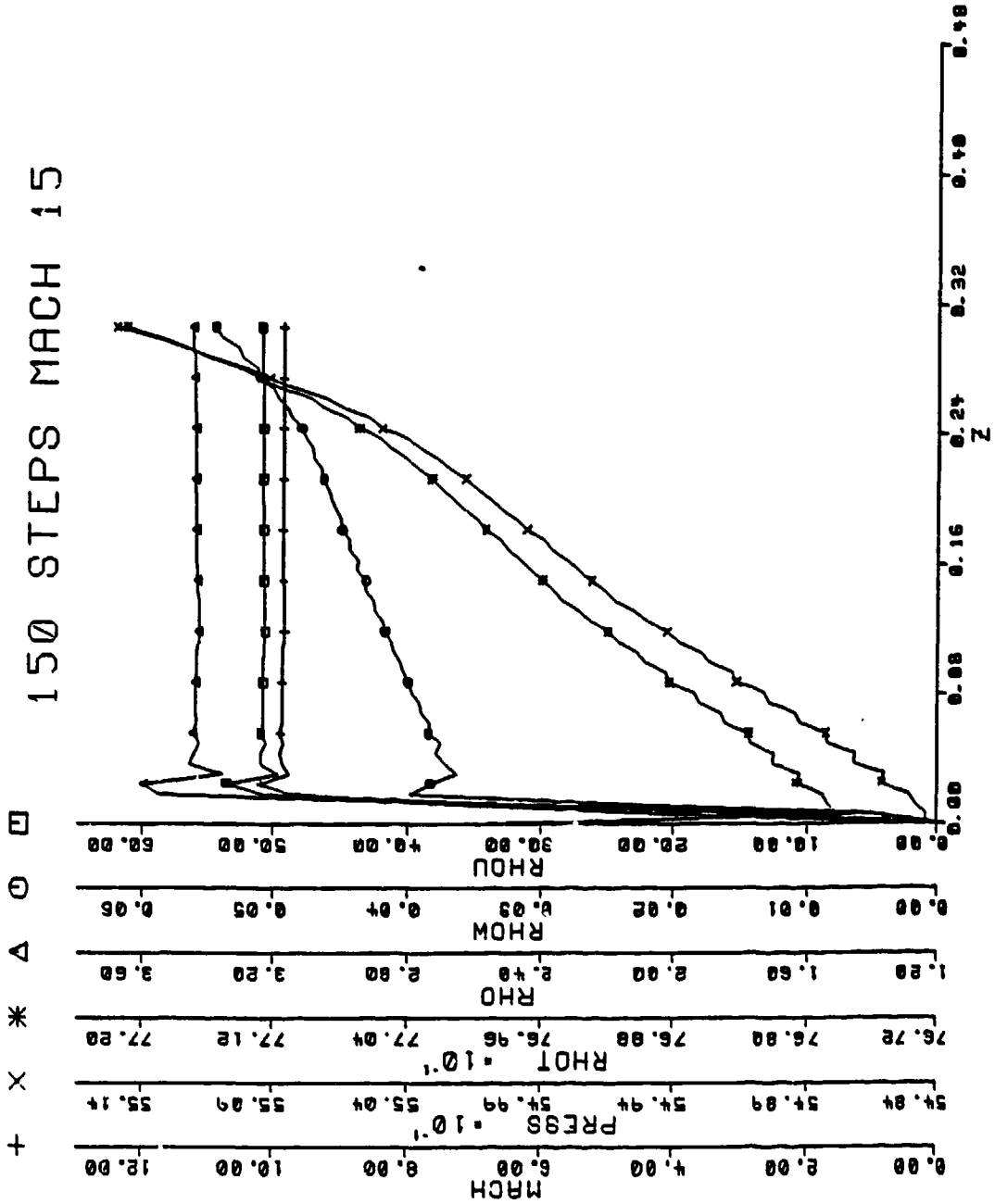


Figure 9.4 State Variable Profiles after 150 Steps at Mach 15, $\Delta x = 0.03$

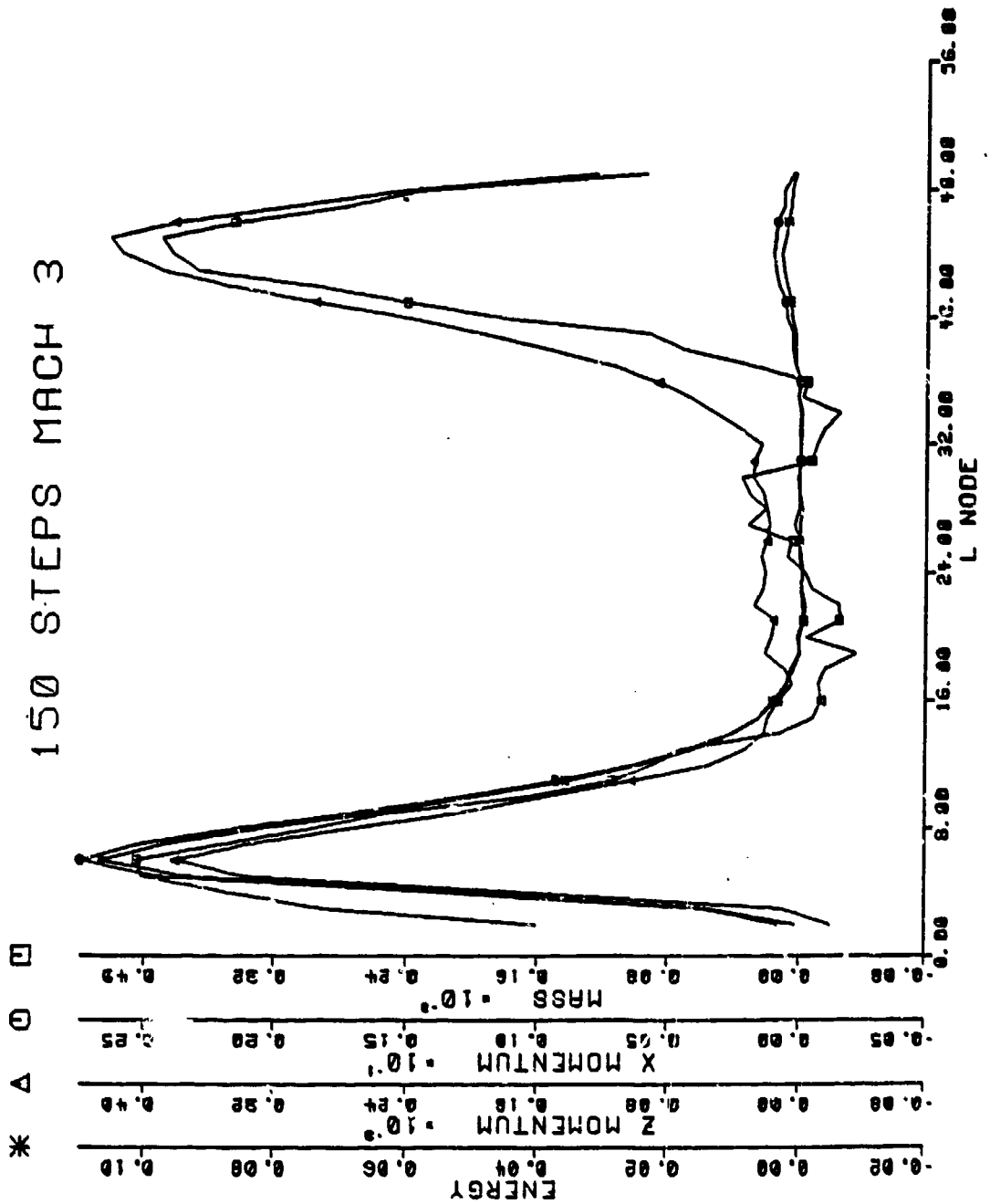


Figure 9.5 Values of Governing Equations after 150 Steps at Mach 3, $\Delta x = 0.03$

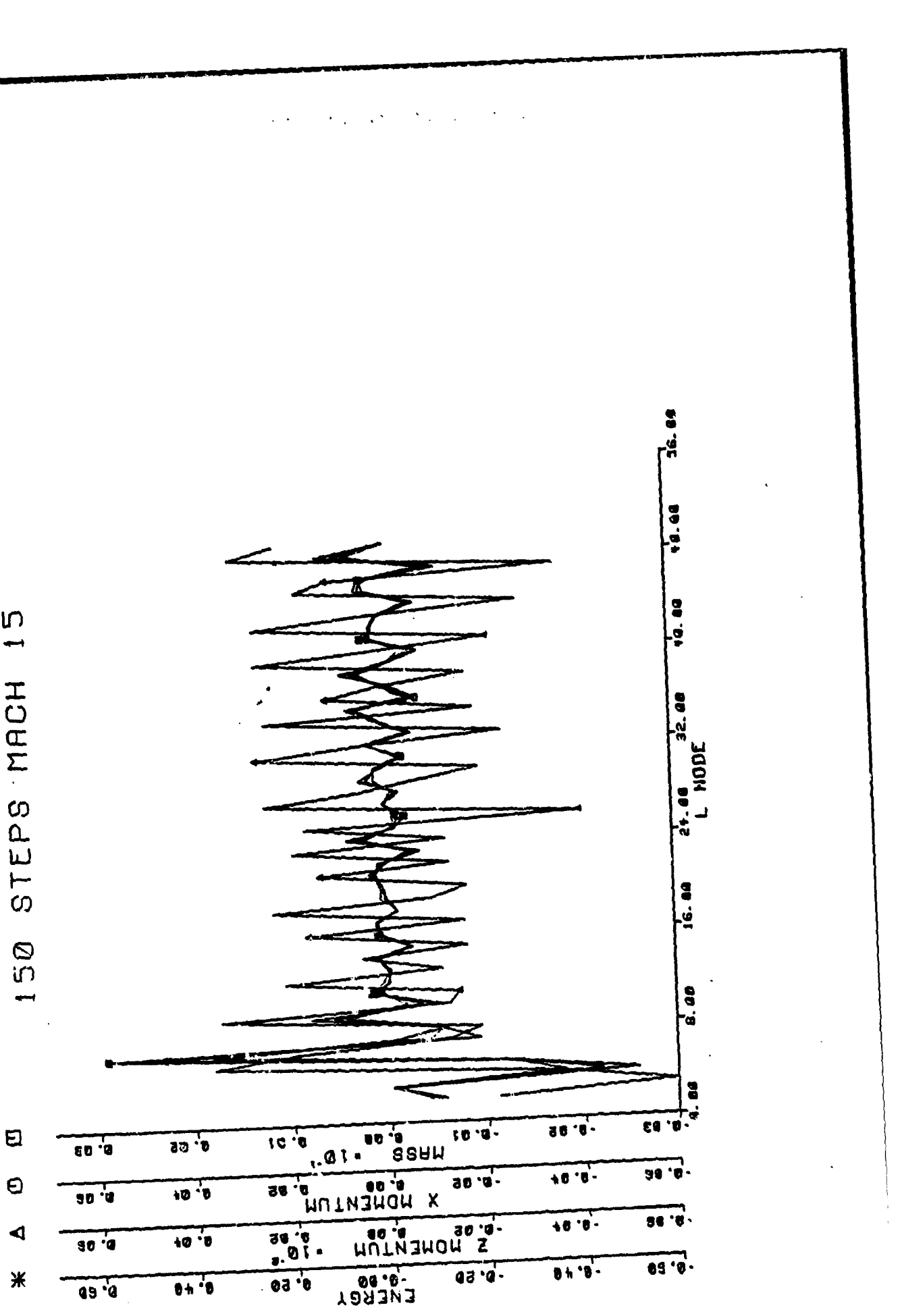


Figure 9.6 Values of Governing Equations after 150 Steps at Mach 15, $\Delta x = 0.03$

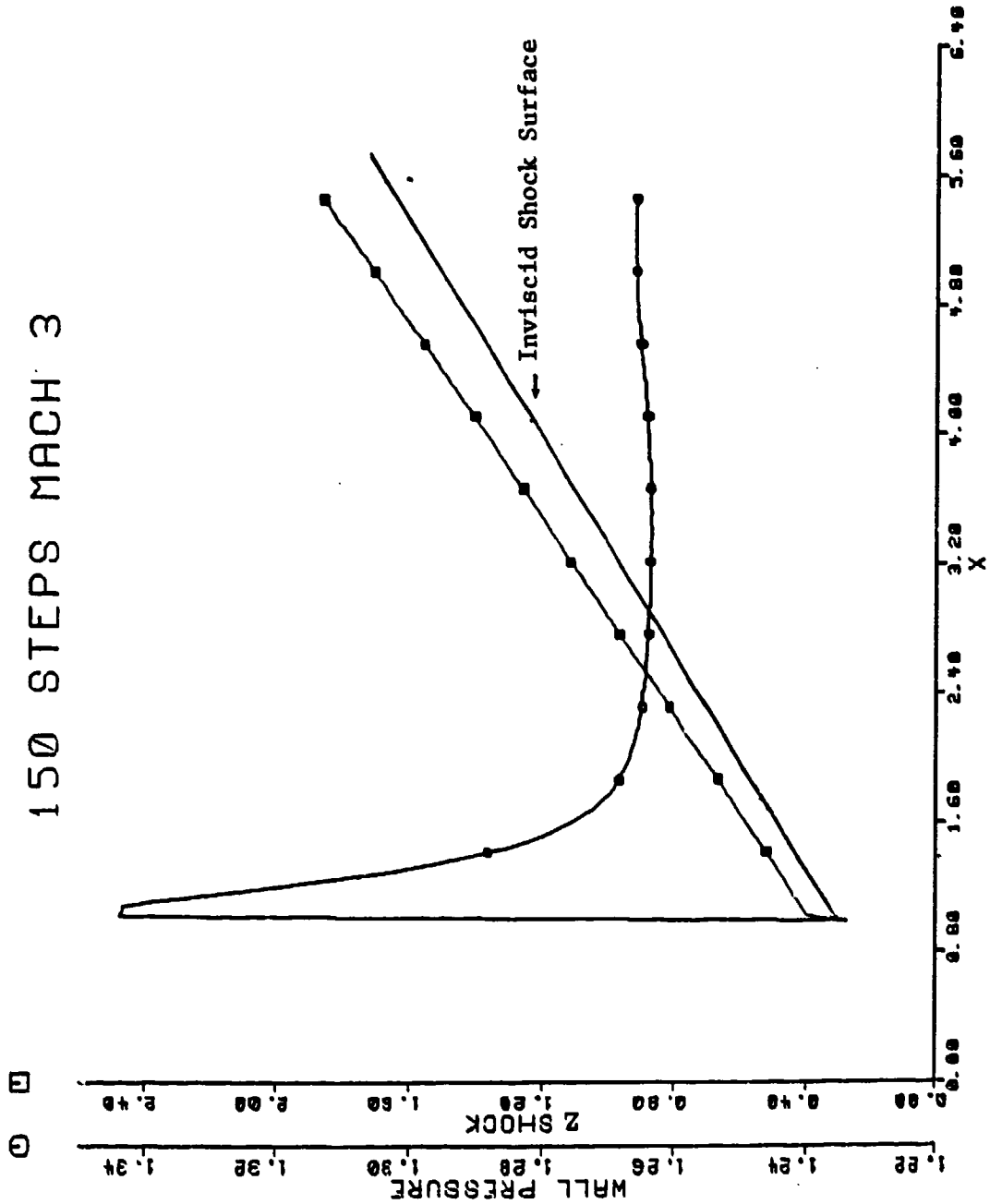


Figure 9.7 Wall Pressure and Shock Surface after 150 Steps at Mach 3, $\Delta x = 0.03$

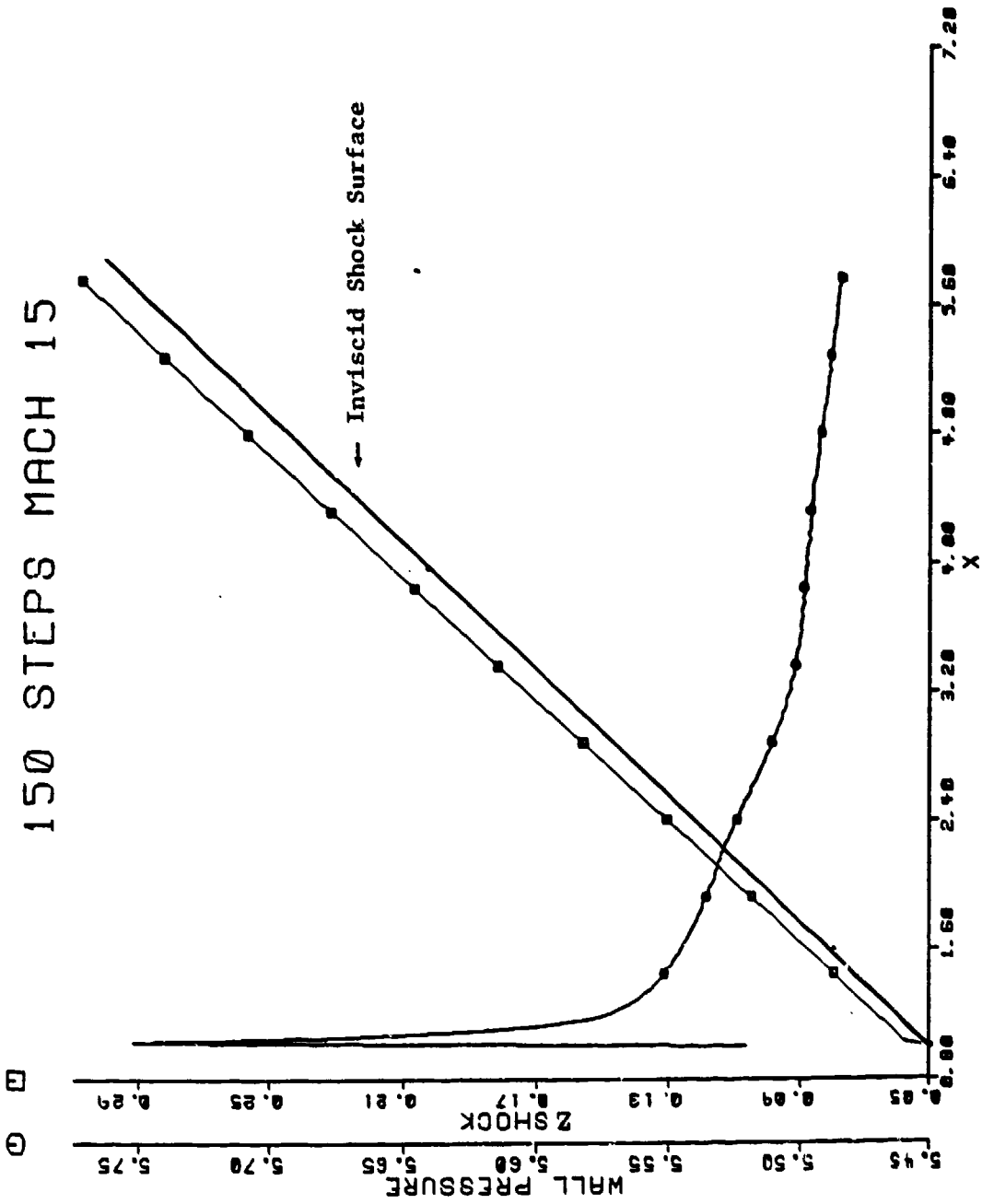


Figure 9.8 Wall Pressure and Shock Surface after 150 Steps at Mach 15, $\Delta x = 0.03$

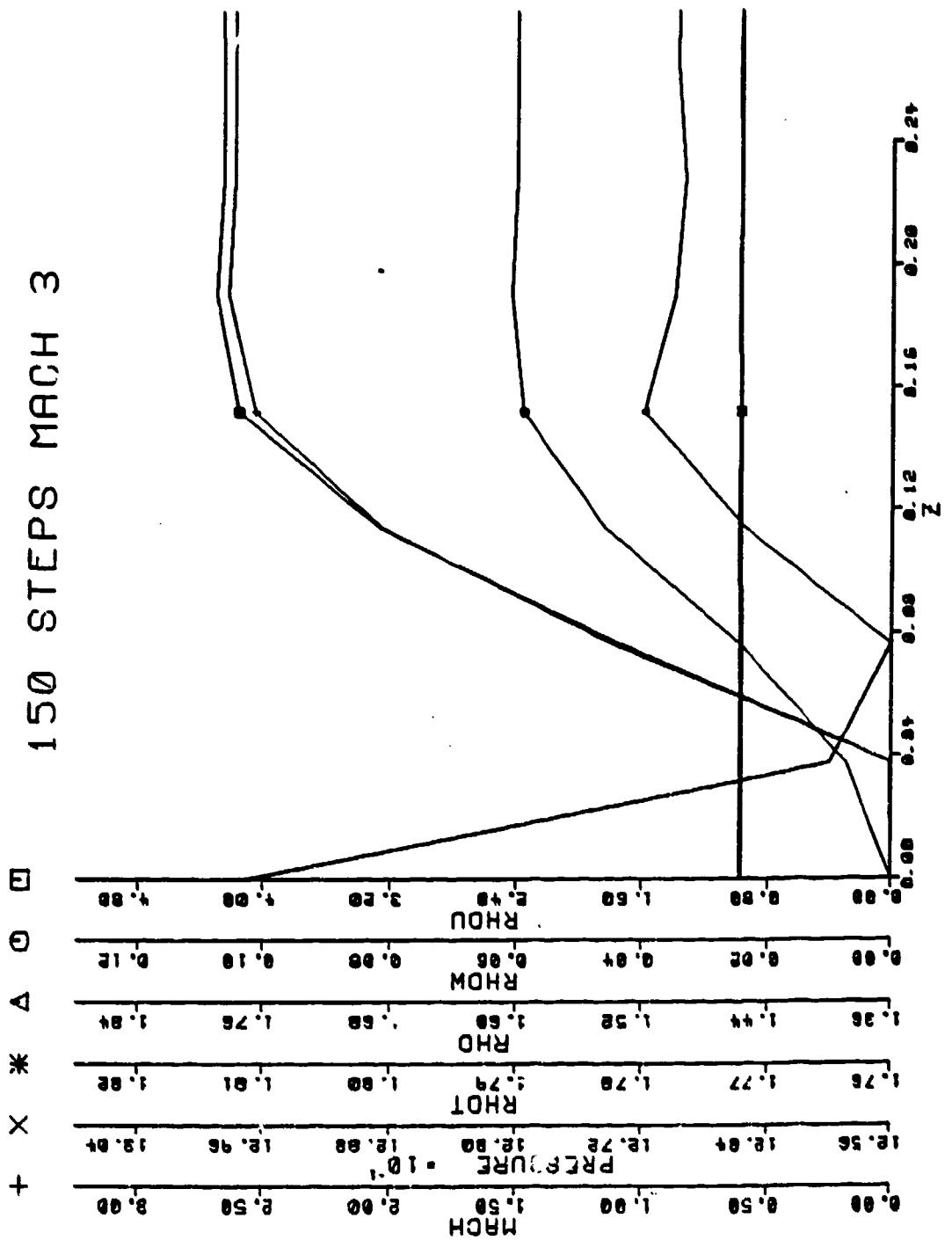


Figure 9.9 Boundary Layer after 150 Steps at Mach 3, $\Delta x = 0.03$

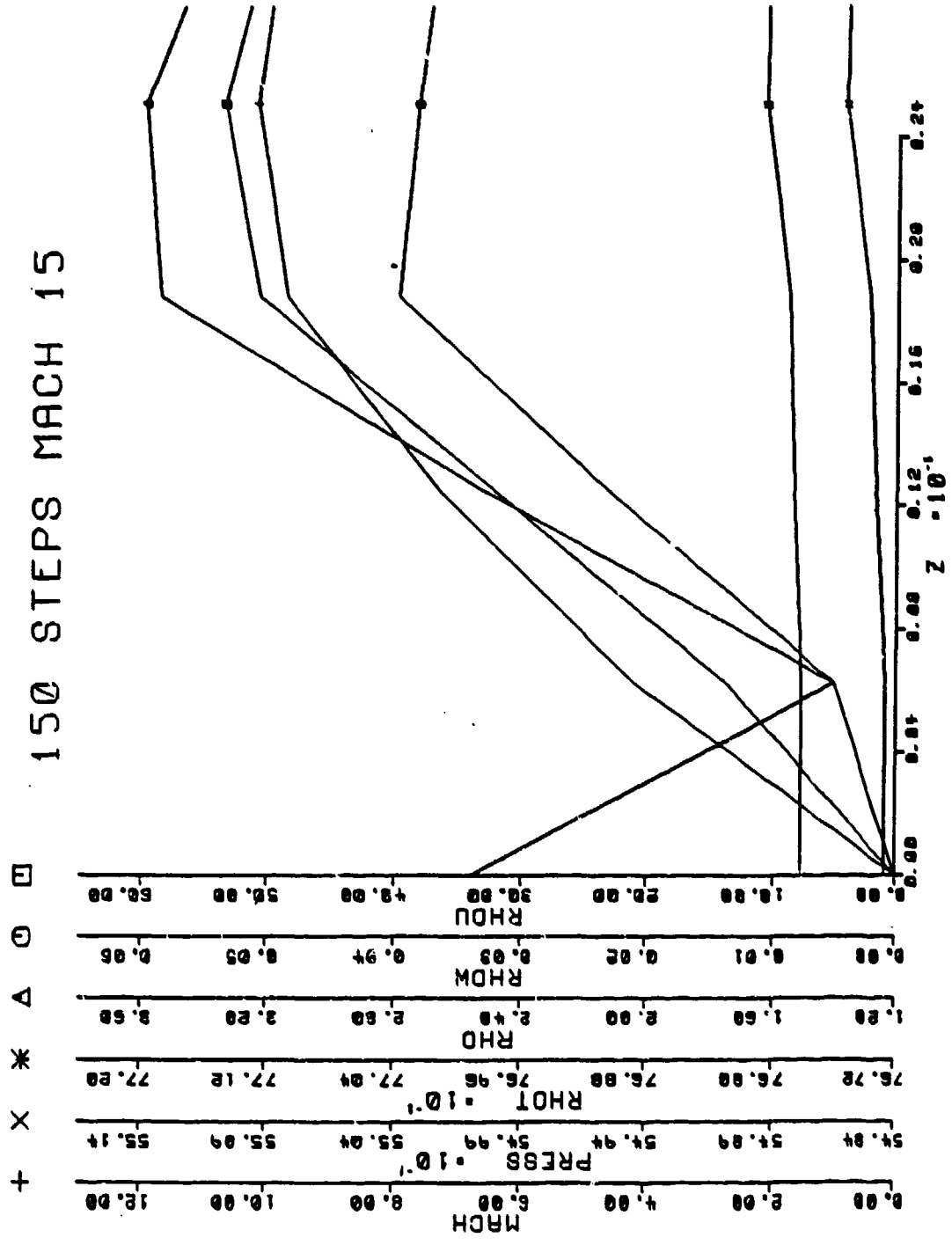


Figure 9.10 Boundary Layer after 150 Steps at Mach 15, $\Delta x = 0.03$

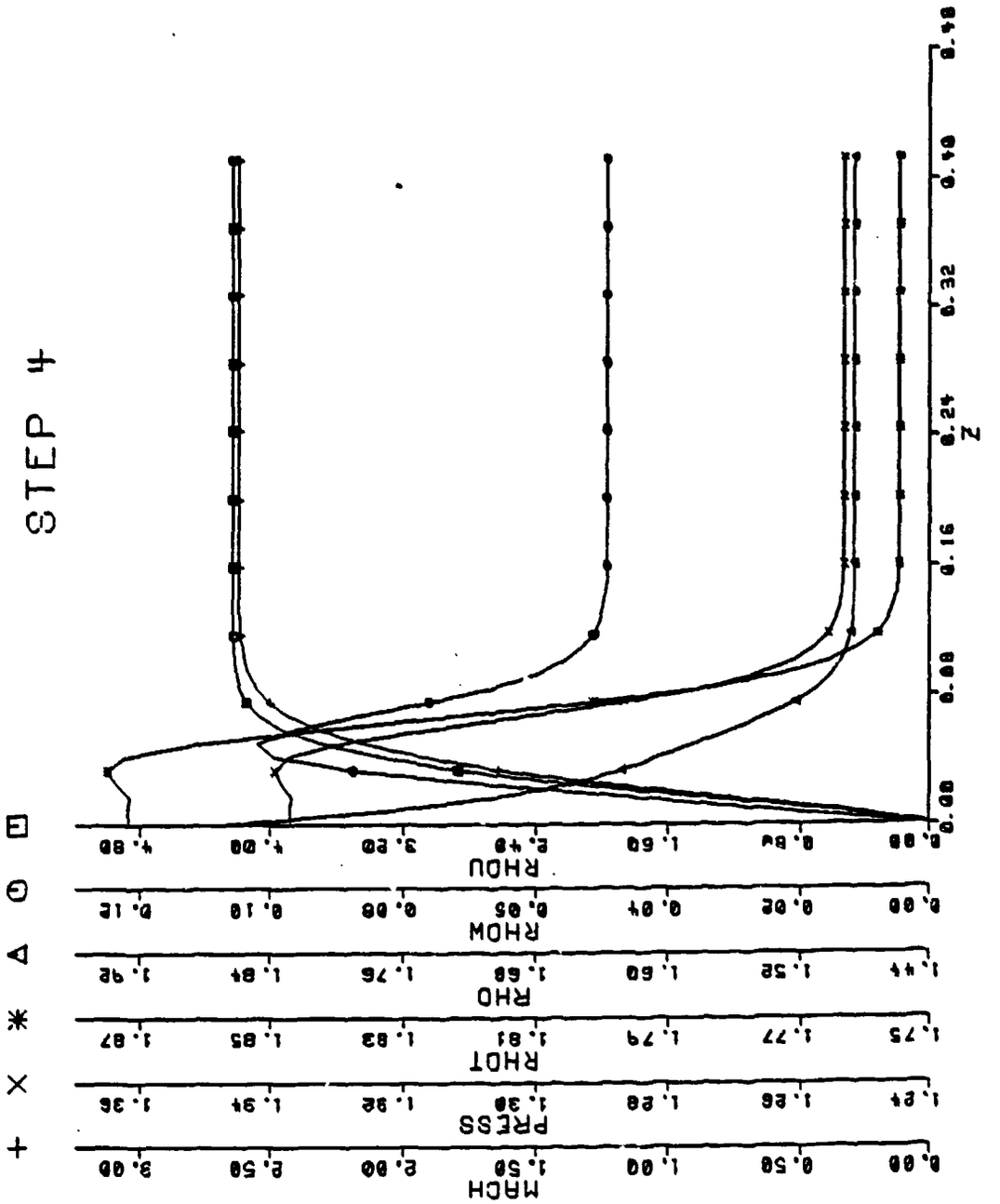


Figure 9.11 Case 4 Profiles, X = 1.12

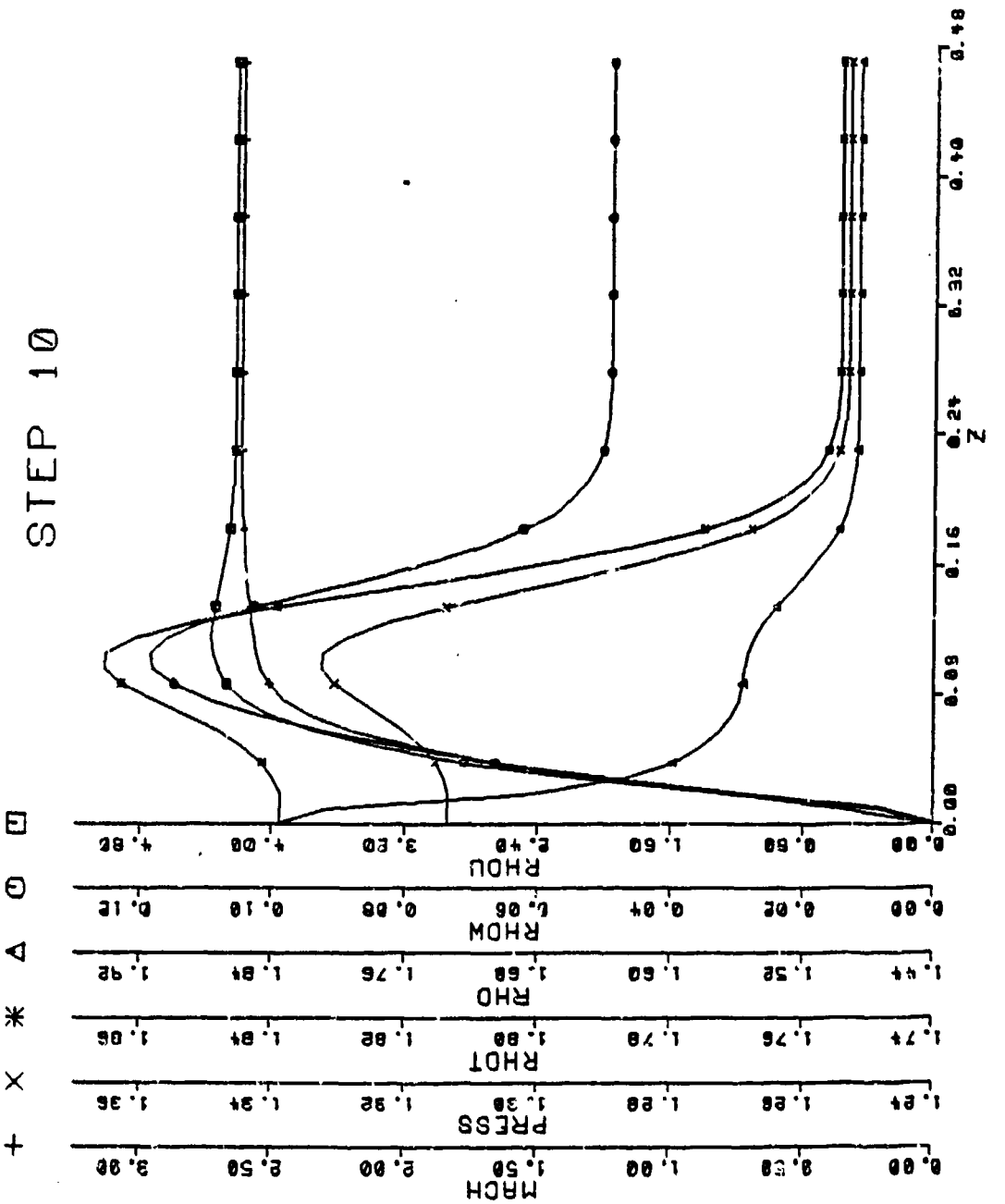


Figure 9.12 Case 4 Profiles, X = 1.30

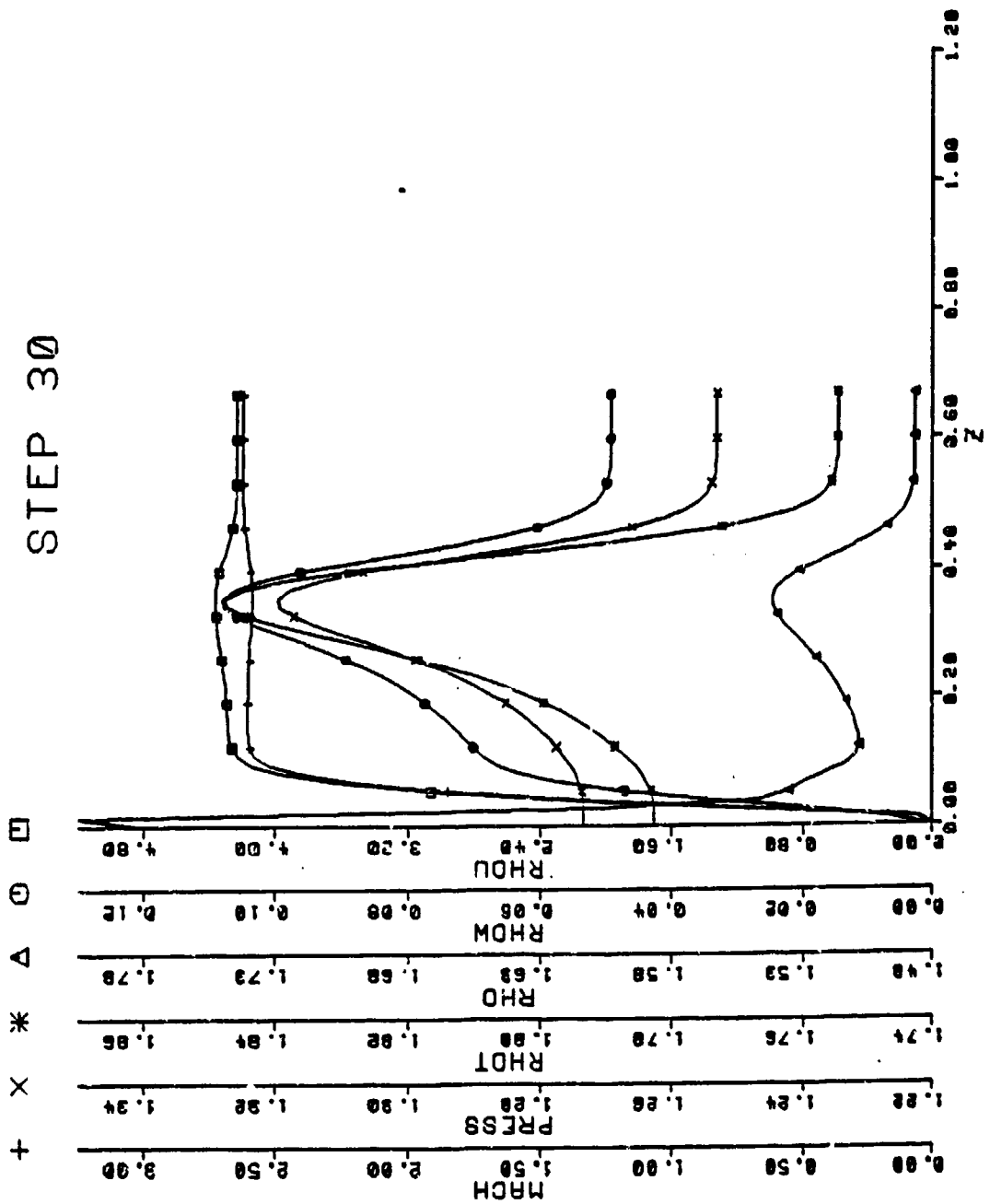


Figure 9.13 Case 4 Profiles, X = 1.90

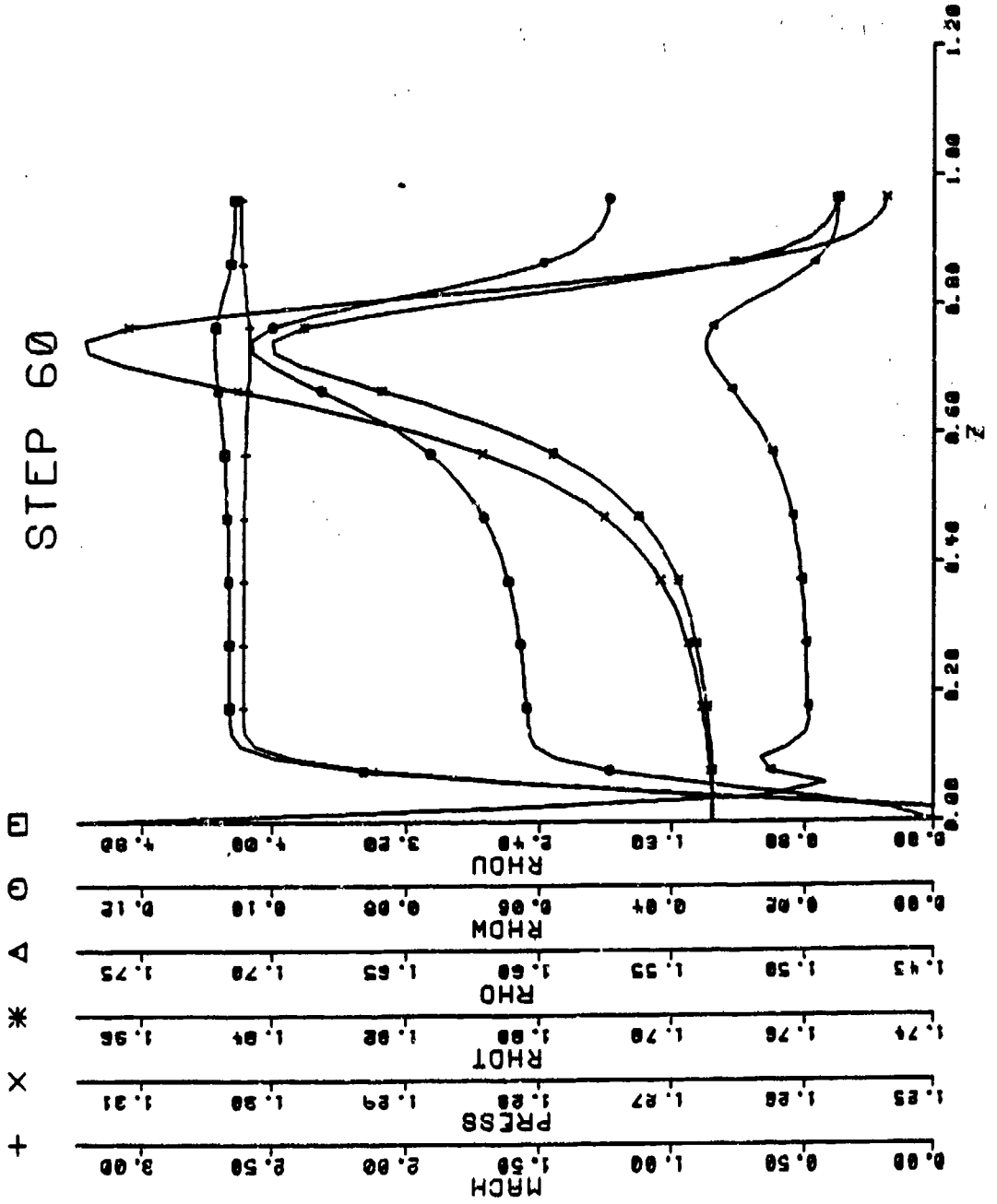


Figure 9.14 Case 4 Profiles, X = 2.80

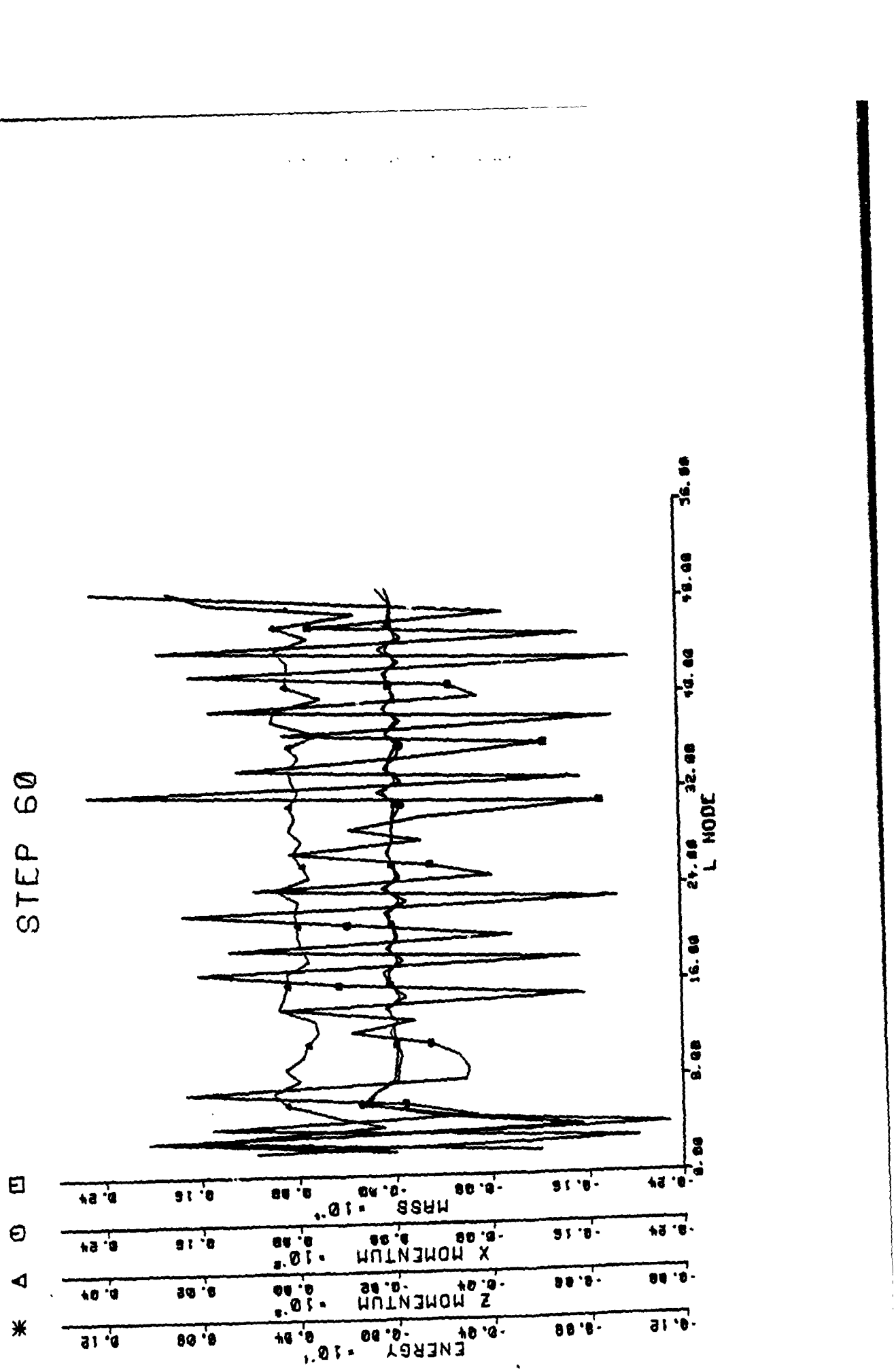


Figure 9.15 Values of Governing Equations of Case 4, X = 2.80

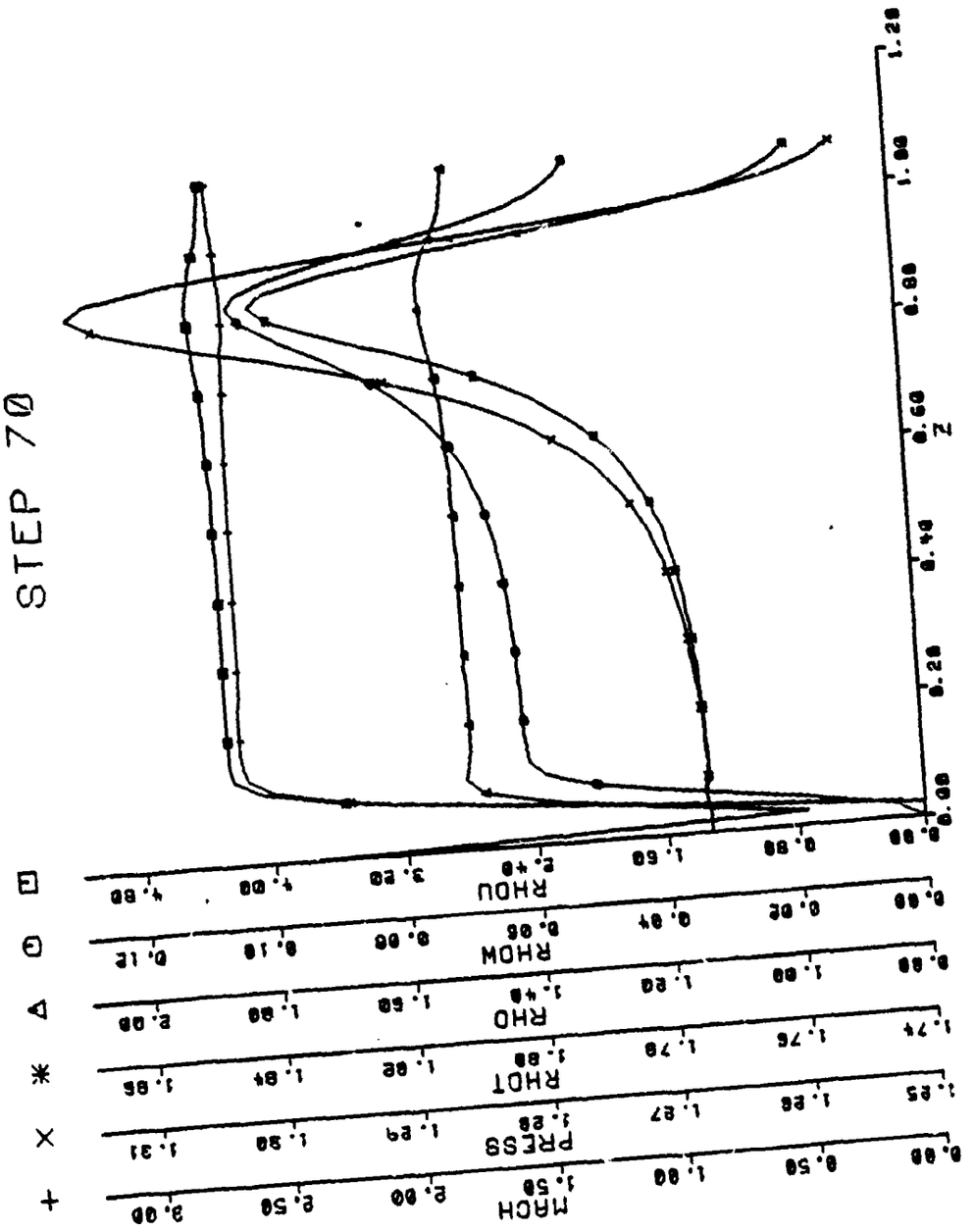


Figure 9.16 Case 4 Profiles, X = 3.10

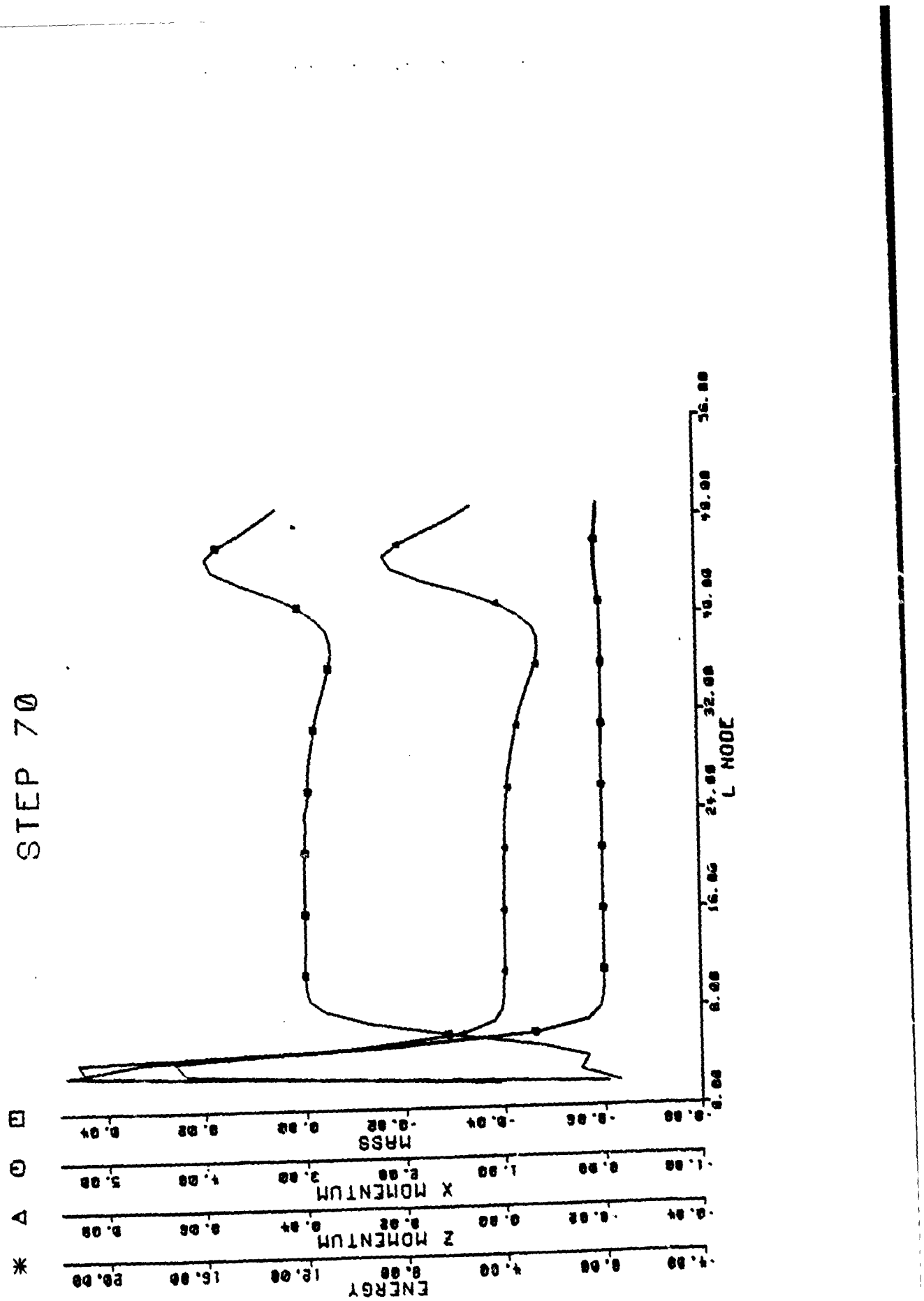


Figure 9.17 Values of Governing Equations of Case 4, X = 3.10

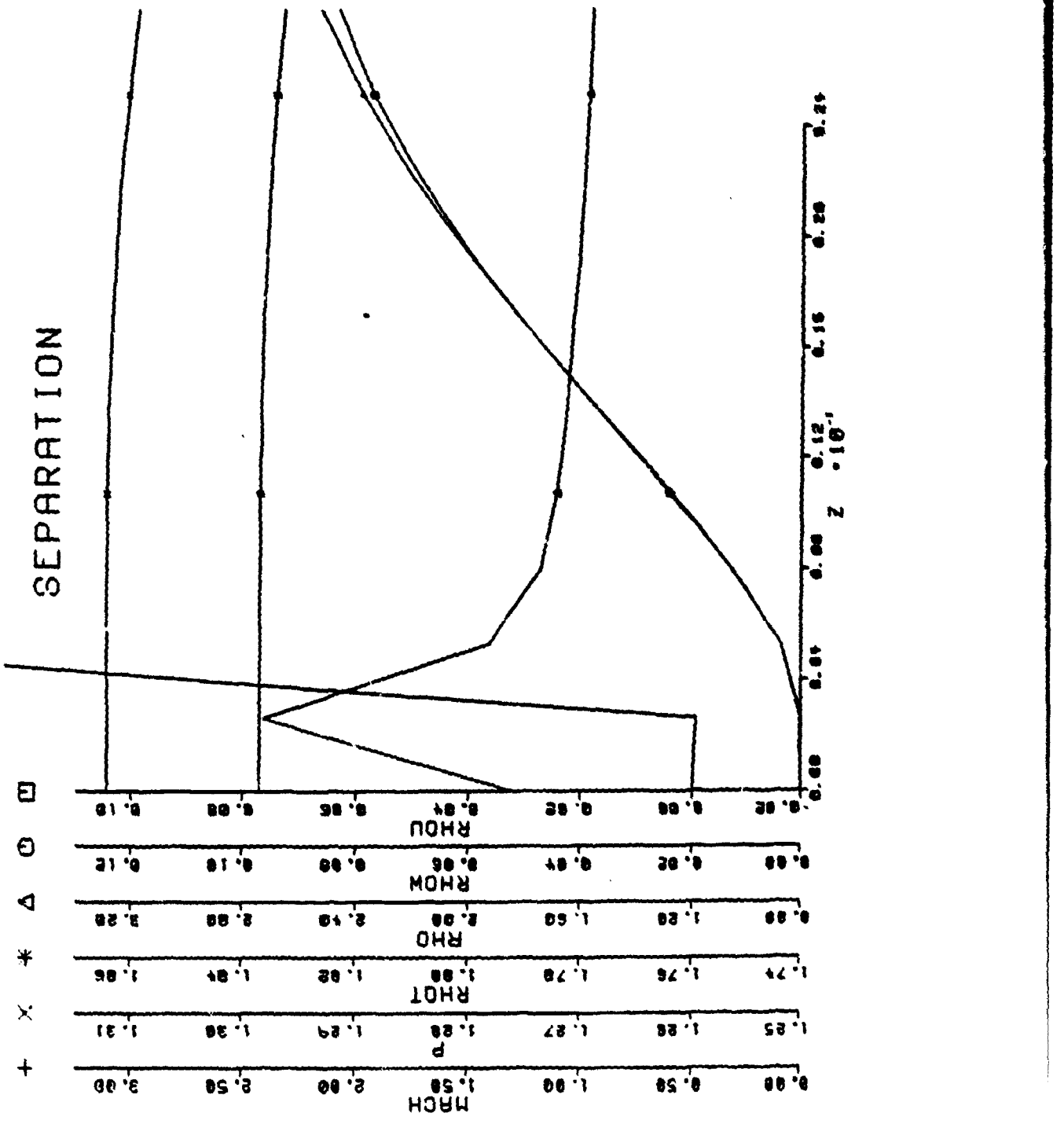


Figure 9.17a Onset of Separated Flow, Case 20

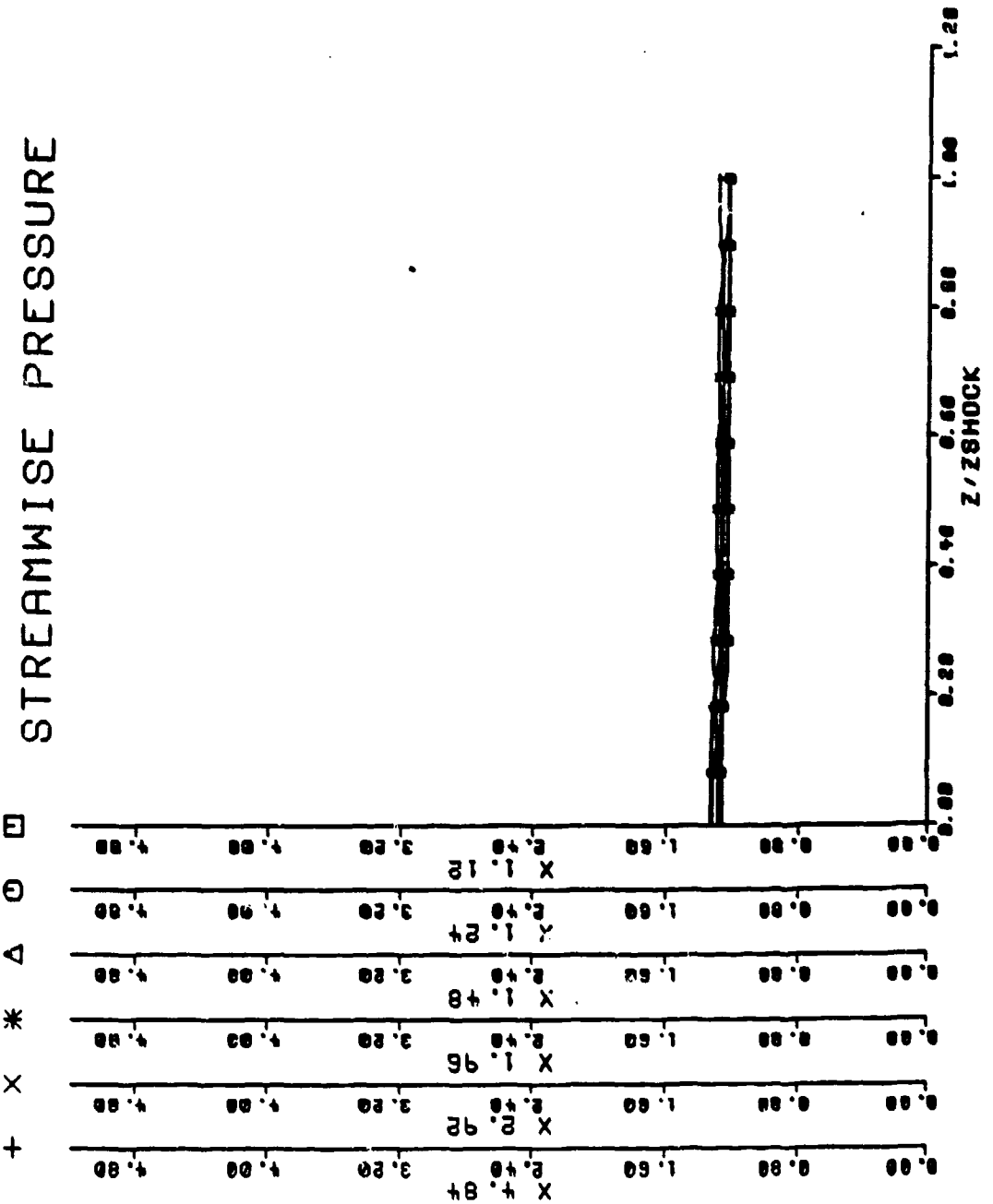


Figure 9.18 Streamwise Pressure Distribution of Case 9

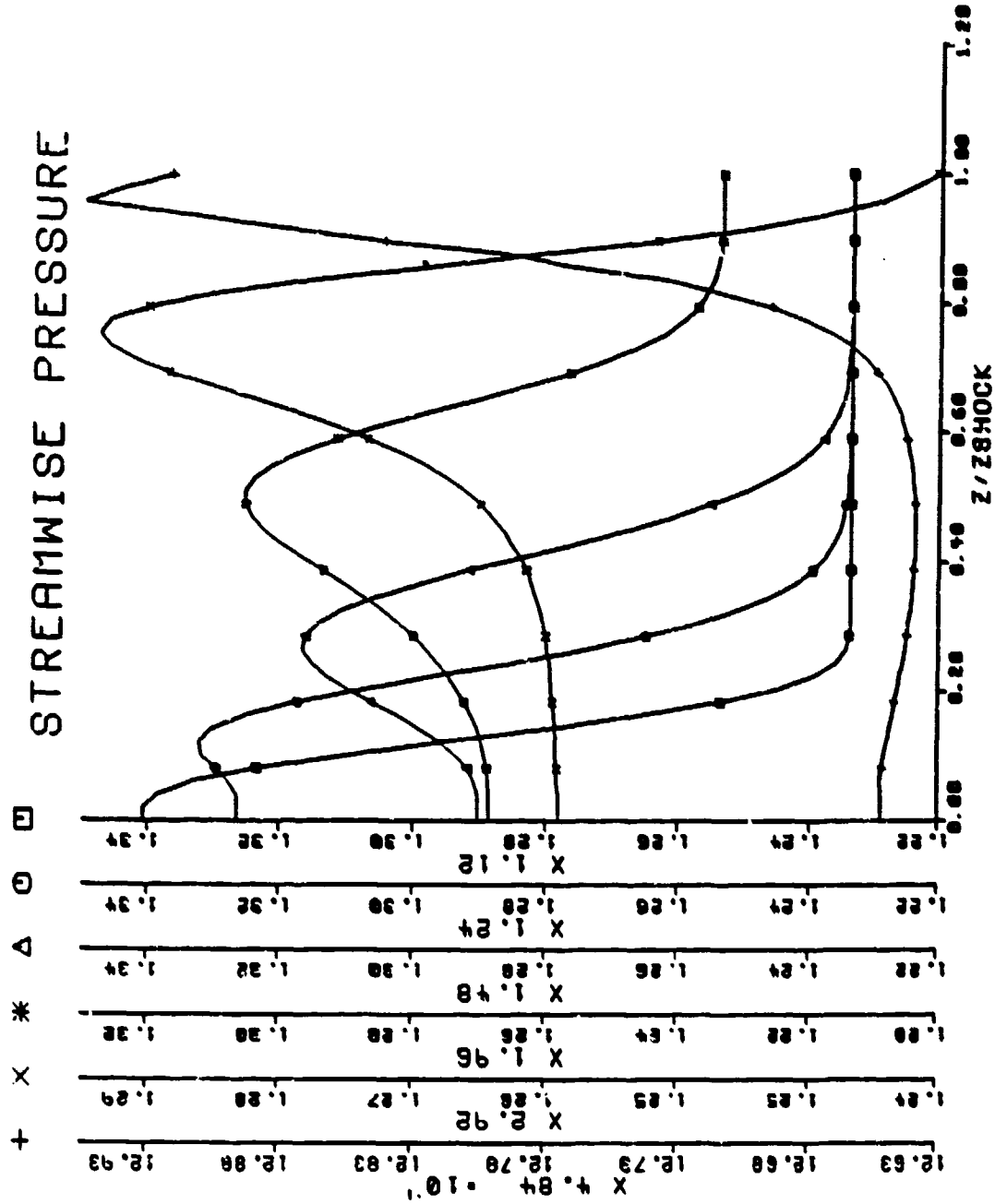


Figure 9.18a Blowup of Case 9 Streamwise Pressure

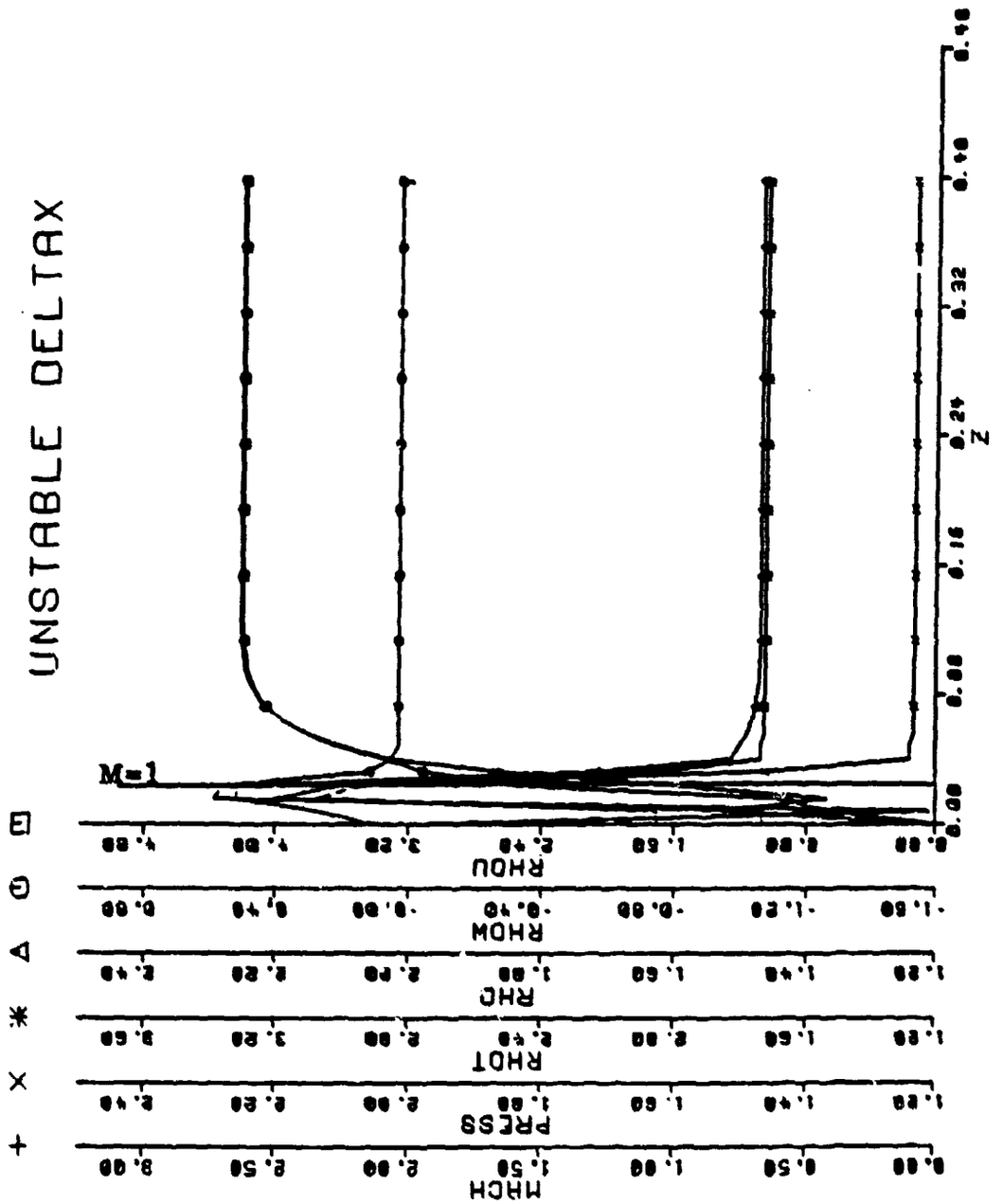


Figure 9.19 Case 6 Profiles, X = 1.017

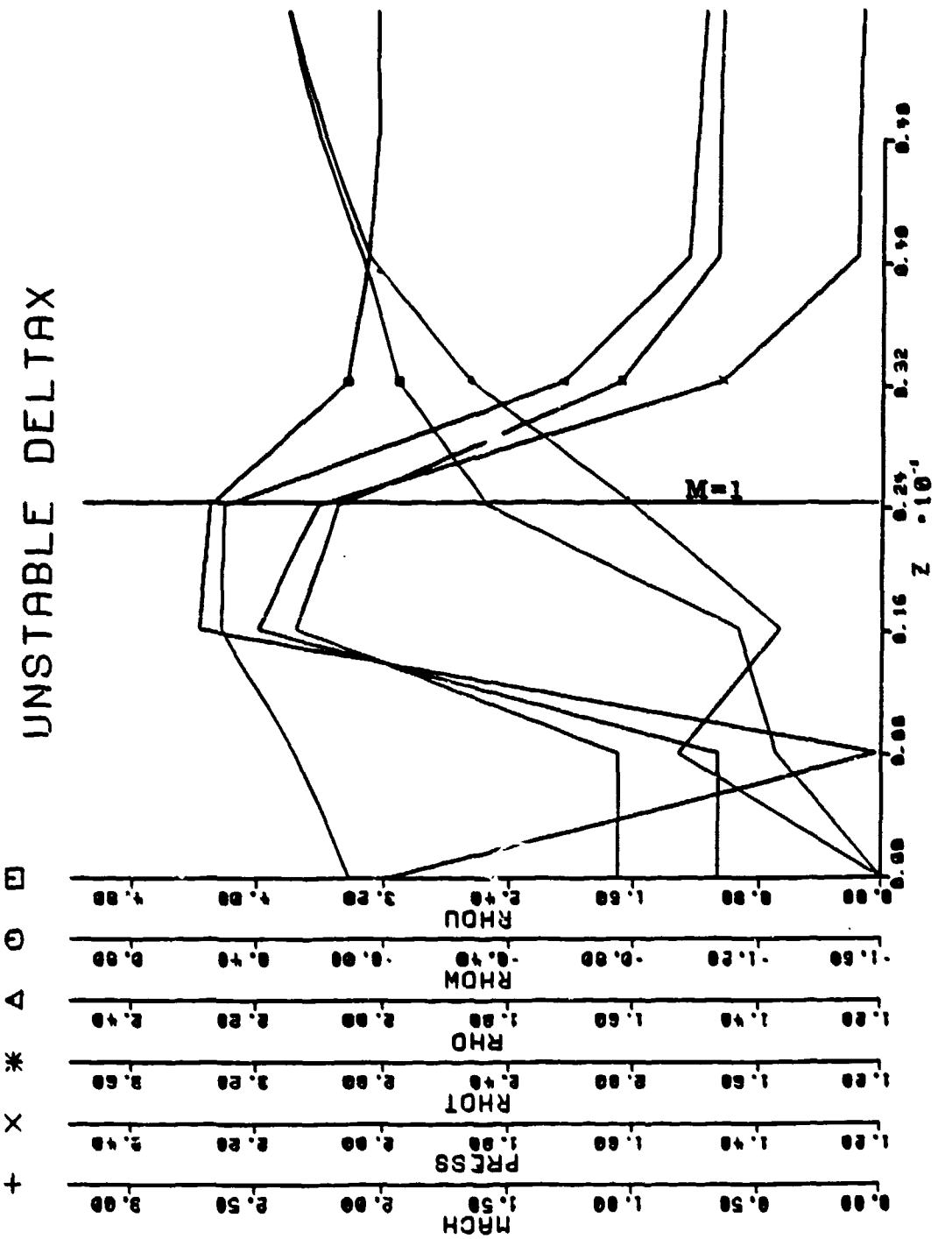


Figure 9.19a Blowup of Case 6 Profiles

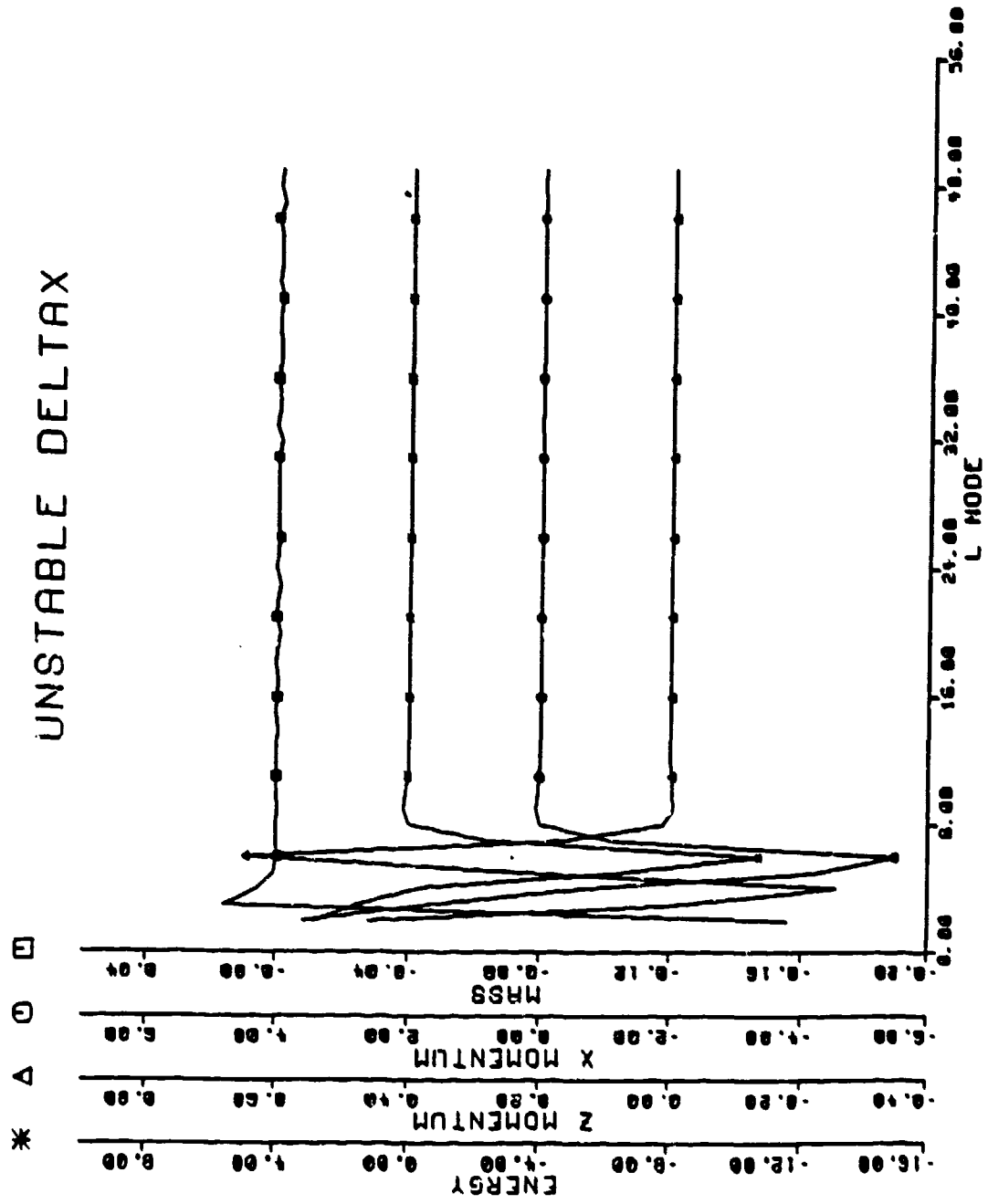


Figure 9.20 Values of Governing Equations for Case 6

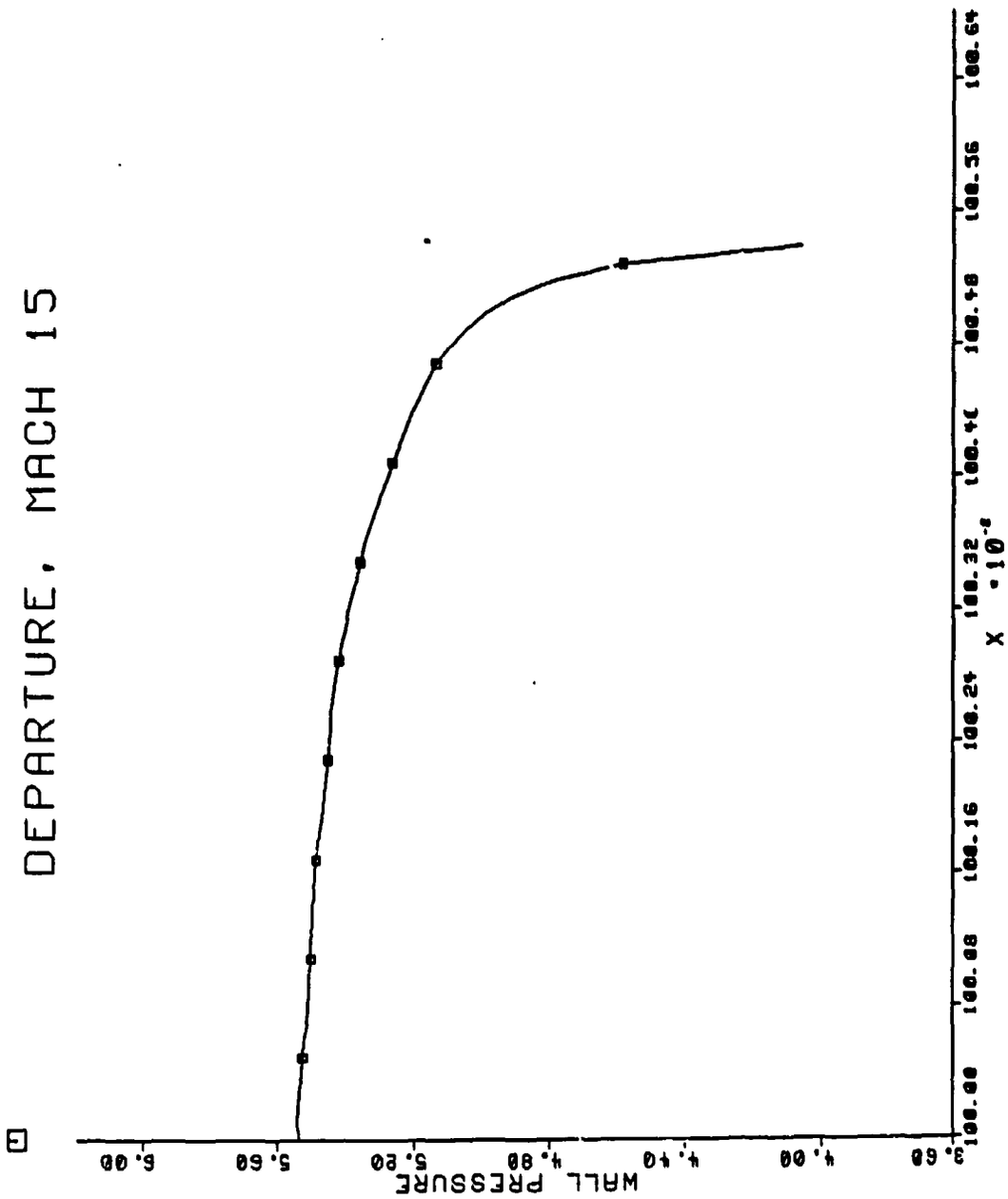


Figure 9.21 Departure Wall Pressure, Case 24

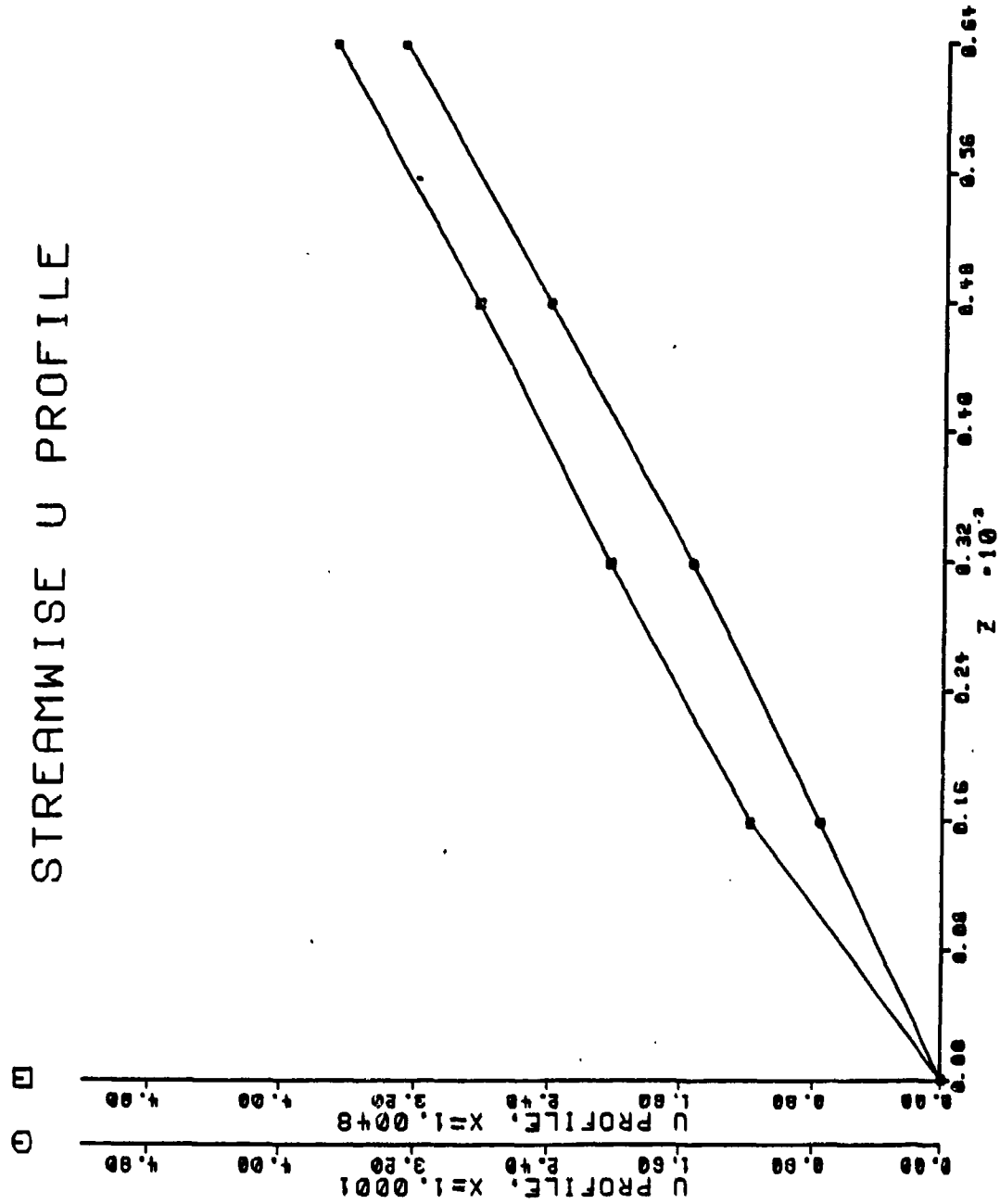


Figure 9.22 Departure Velocity Profiles, Case 24

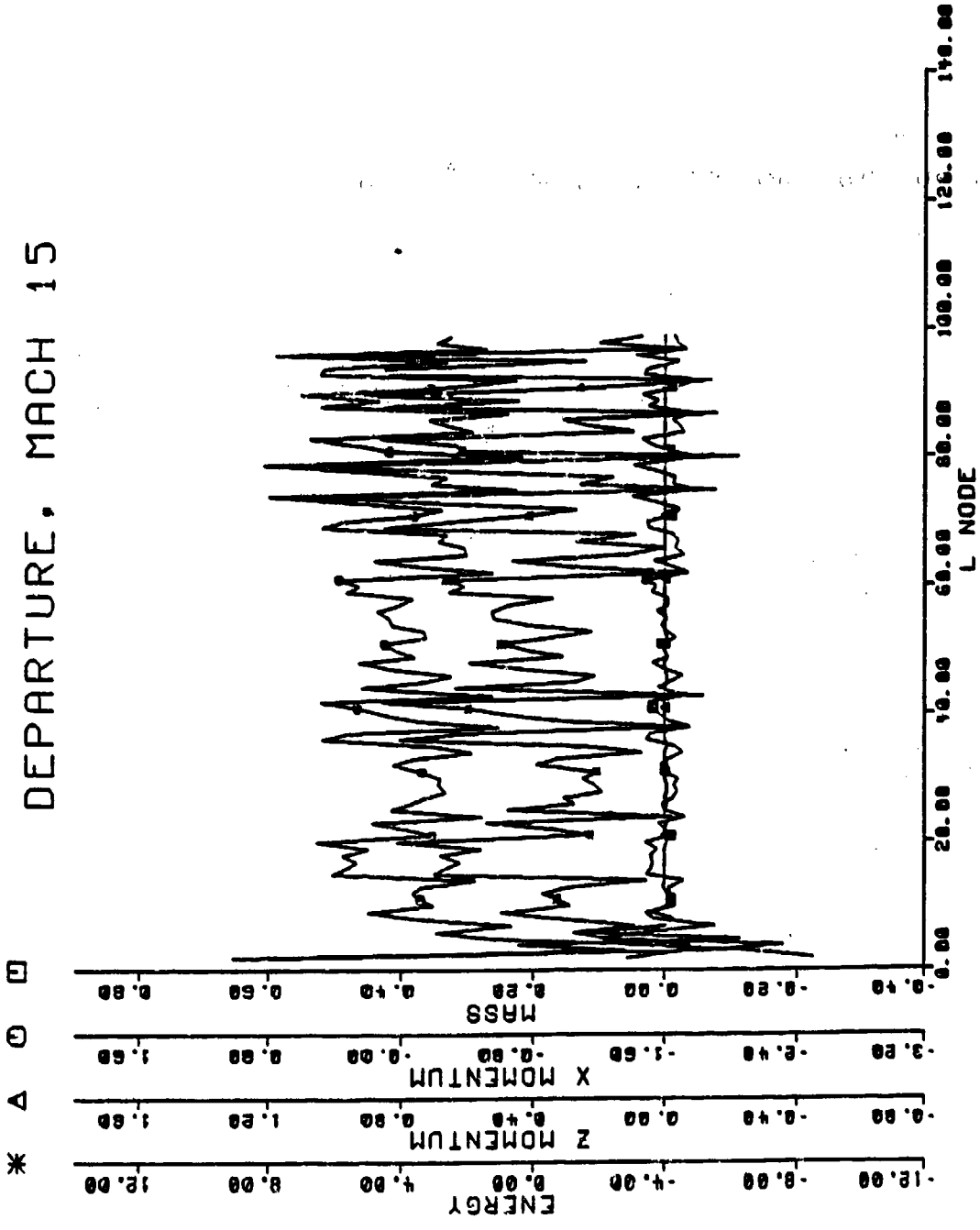


Figure 9.23 Values of Governing Equations at Departure, Case 24

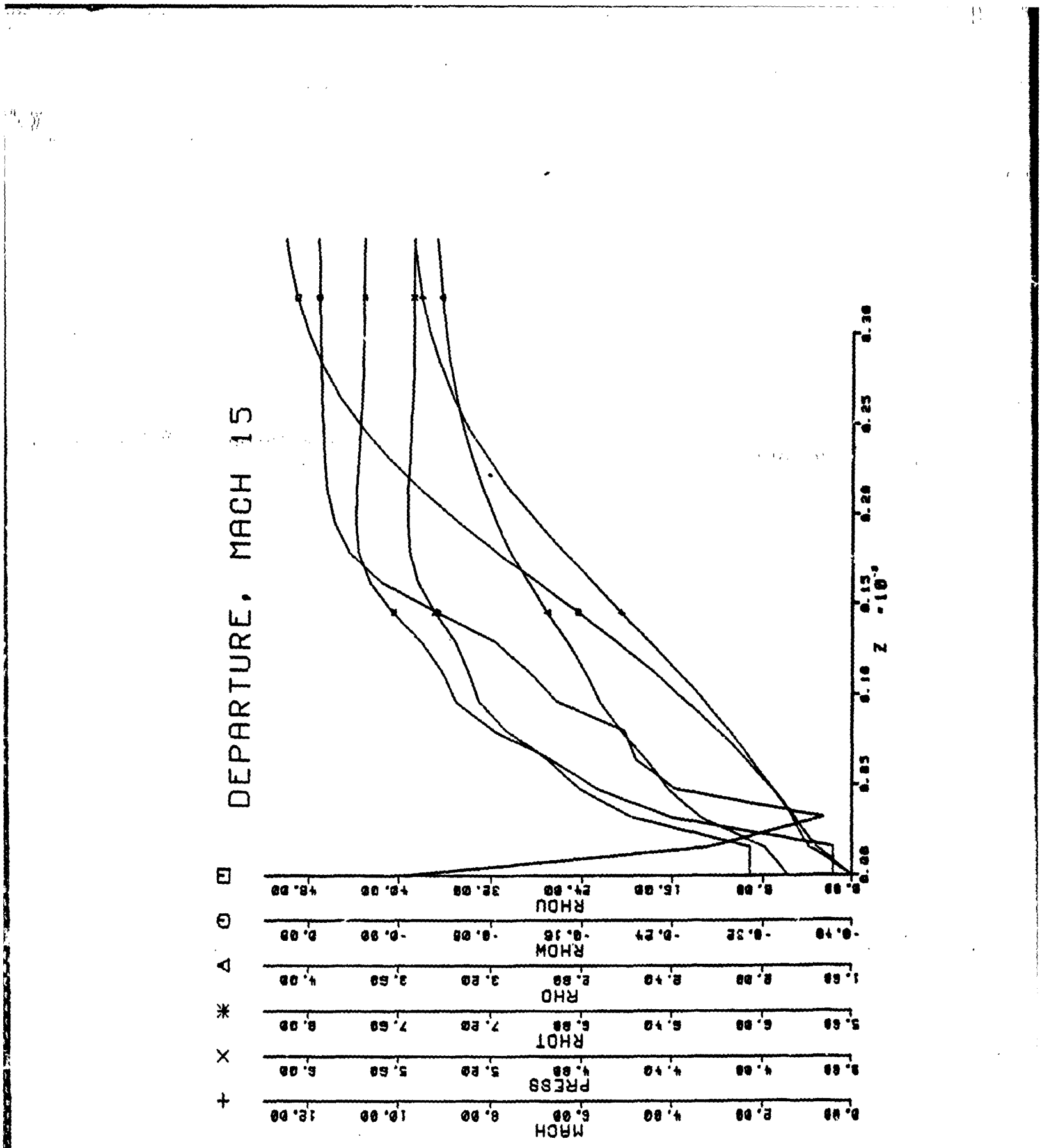


Figure 9.24 Departure Boundary Layer Profiles, Case 24

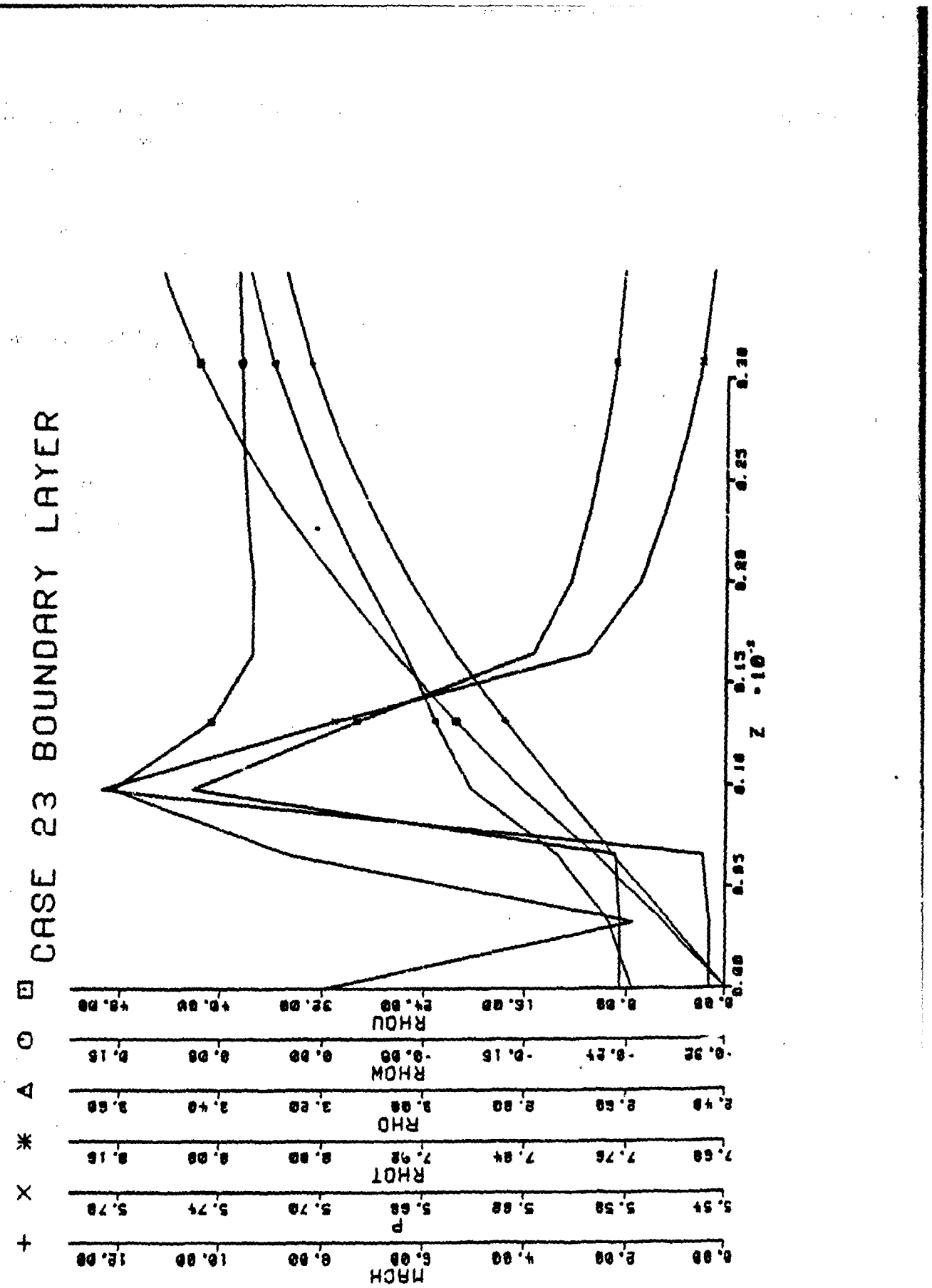


Figure 9.25 Boundary Layer Profiles, Case 23

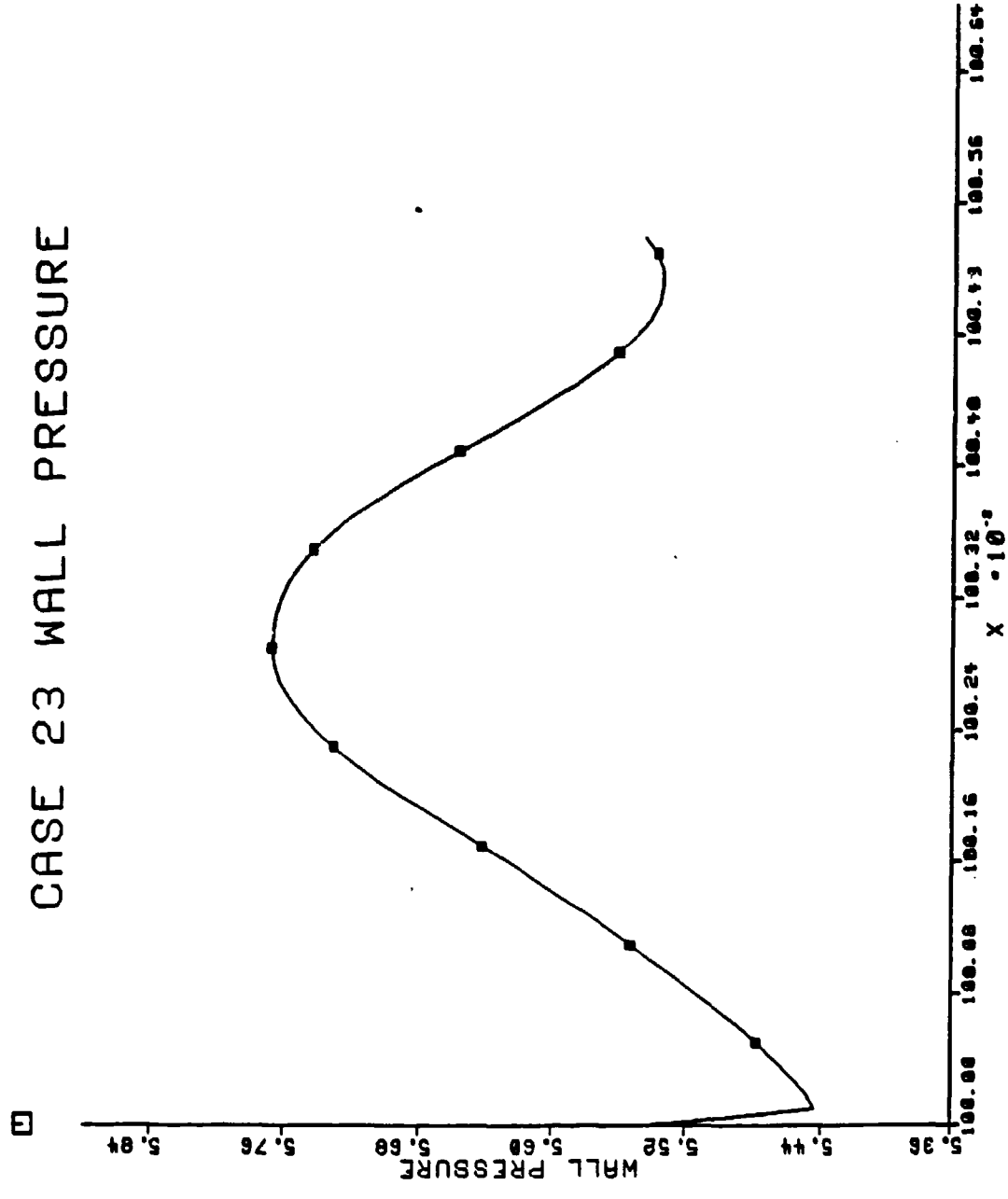


Figure 9.26 Wall Pressure, Case 23

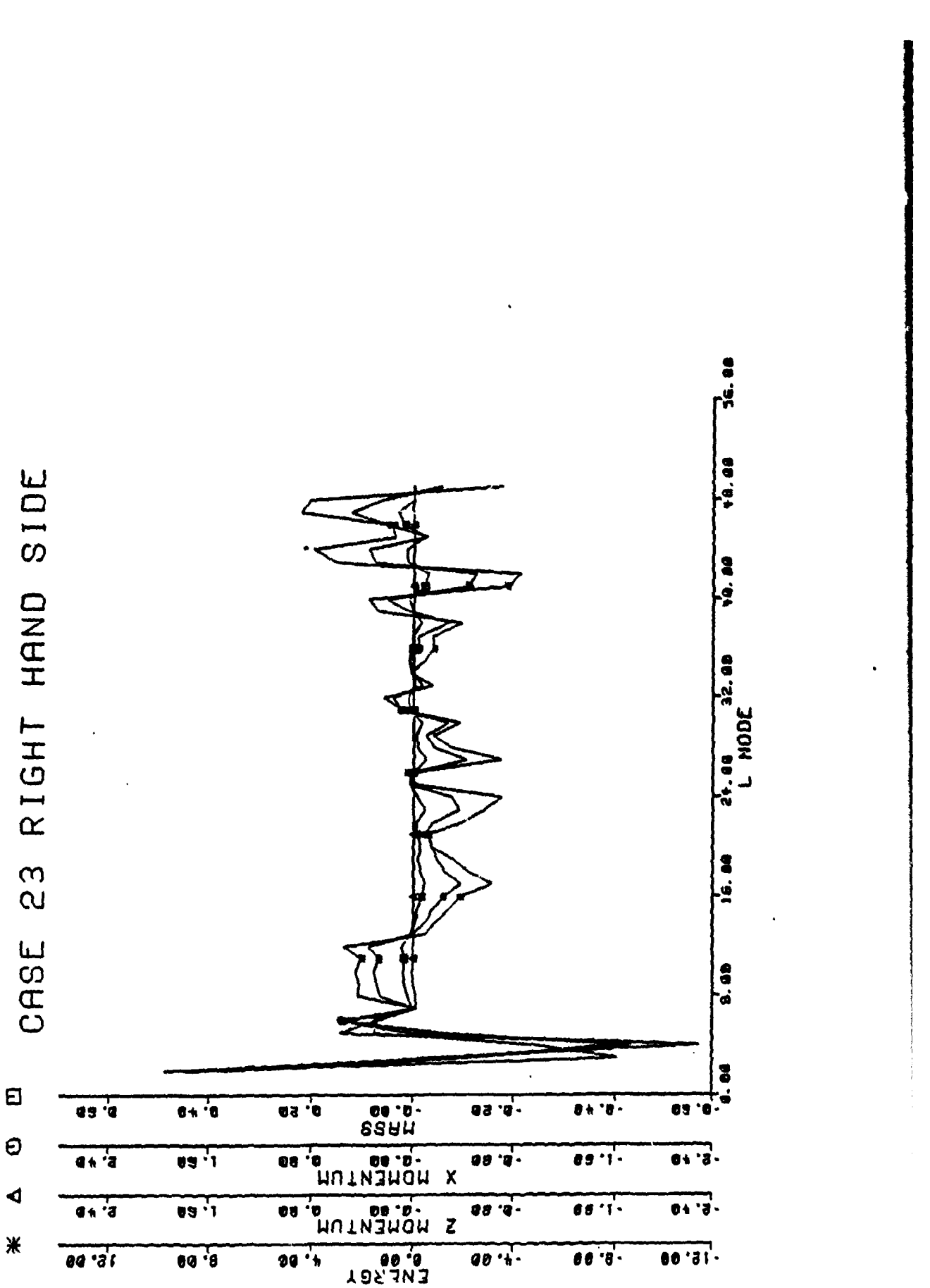


Figure 9.27 Values of Governing Equations, Case 23

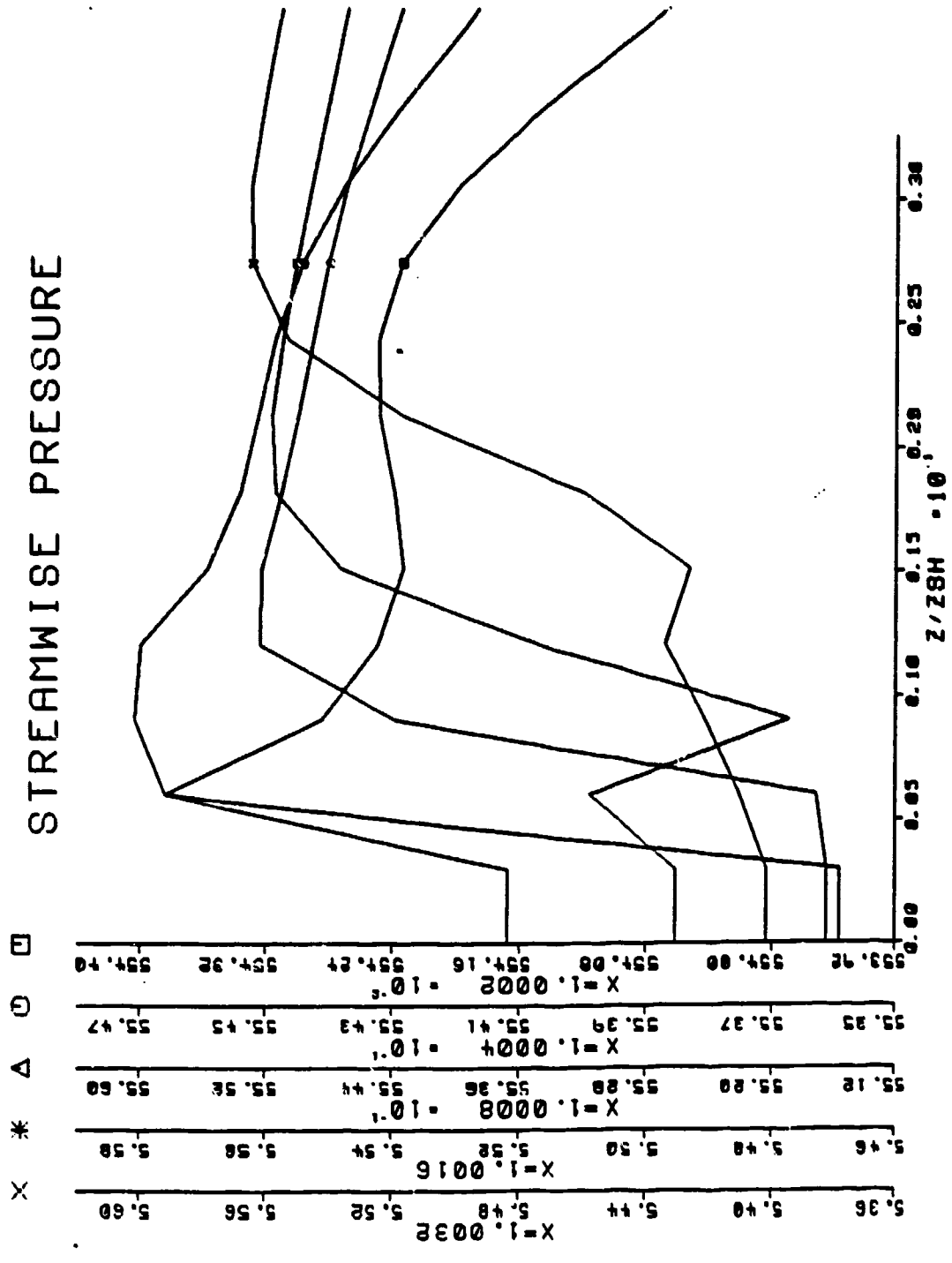


Figure 9.28 Streamwise Pressure During Departure, Case 24

10.0 CONCLUSIONS

The Bhutta-Lewis Algorithm (Ref. 1) behaves in an expected way for PNS schemes that do not make specific sublayer assumptions. Of particular relevance to this work in reaching this conclusion are References 13, 14 and 15. The latter specifically address the role of PNS modeling and related questions of marching, departure behavior and stability for PNS equations. They particularly discuss in some depth the use of the streamwise step size to overcome weak ellipticity. The present analysis uses step size as a way to characterize the behavior of the Bhutta-Lewis' approach.

The results of Chapter 9 indicate that the algorithm converges to and breaks down by approaching a separated flow condition. This is in agreement with the results of the stability analyses of Chapter 8. Exponential departure behavior may not be apparent unless the grid and stepsize are sufficiently small, but the departure behavior is manifested by convergence to separated flow.

Recall from Chapter 5 that the tridiagonal algorithm procedure is associated with pseudo-temporal steps (iterations) rather than space. Therefore the stability analyses of References 13, 14 and 15 which involve implicit space steps are not applicable. However, it is reasonable to assume that the

current iterative approach amounts to taking the same step size repeatedly and step size does then characterize the behavior.

A minimum streamwise step size is related to the height of the subsonic sublayer.[13,14,18] The relation developed in Reference 13 is:

$$\Delta x/\delta_s > 1/\pi$$

where δ_s is the thickness of the subsonic layer. The larger the elliptic zone, the more restrictive on accuracy the above condition becomes.[13] This explains the results of Case 6. Also, it is implicit that stable solutions are more easily achieved with high speed flows in view of their relatively smaller subsonic layers.

This behavior is consistent with the findings described in Chapter 9. The algorithm will march downstream when certain conditions are met. Only if the subsonic layer is sufficiently small, e.g. as in the 4% or 5% boundary layer cases of Tables 9.1 and 9.2, is stability evident. Larger boundary layers can be marched with higher Mach numbers because the subsonic layer remains small. This agrees with References 13 and 14. In fact, when this condition is met, a rather large step size of 0.06 was used successfully. Smaller step sizes of order 10^{-3} or 10^{-4} also work and step sizes of this order have been used by others.[4,9,13,14,18] It must be concluded that the Bhutta-Lewis algorithm (Ref. 1) behaves in the same manner as was to be expected for PNS schemes without sublayer assumptions.

It should be noted that the hypersonic similarity parameter

$$M_\infty/(Re)^{1/2}$$

was the basis here for selecting specific Reynolds numbers (Chapter 9). At Mach numbers 3 or 15, a match of the hypersonic similarity parameter to that of Bhutta and Lewis' original paper (Ref. 1) implies a boundary layer thickness exceeding 20% of the local shock layer thickness. According to the aforementioned stability analysis,[13] this should severely reduce the allowable step size. Indeed, the current research failed to march low Reynolds number, viscous flows with large boundary layers any appreciable distances downstream. Bhutta and Lewis, however, developed the current algorithm to solve low Reynolds number flows and were apparently able to march large distances downstream.

One portion of the Bhutta and Lewis algorithm which the current work did not duplicate was variable step size. In Reference 1, results indicate that the step size begins with $O(10^{-8})$ and ends with $O(1)$. Other documented parameters of the Bhutta and Lewis algorithm have been duplicated and tested and have been shown to not ensure marching stability. Since the current work agrees with past references on the importance of streamwise step size, it may be that variable step size enabled the instabilities to be supressed. This is consistent with the fact that successful marching was indicated with high Reynolds

number flows where the boundary layer and subsonic layer thicknesses do not grow appreciably with downstream distance. As the boundary layer and subsonic layer grow in a viscous flow, the step size apparently must increase to maintain stability, as implied in Reference 1.

Nevertheless, the scheme has been shown to be unstable. Therefore, in order to march successfully, some parameter or parameters in this scheme must be adjusted to allow for marching stability. Unfortunately, the parameters which allow marching stability were not discussed in Reference 1 and no reference was made to the characteristic instability of the scheme.

The role of the differential modification of the equation of state is also interesting. Recall that the modified equation of state:

$$\gamma p + \theta(P,\xi + P,\zeta) = \rho T$$

was used to replace the true equation of state. For most calculations θ was set identically equal to zero; in certain cases, θ was allowed to vary. In Reference 1, Bhutta and Lewis indicate that θ was not used for actual numerical cases, but in Reference 11 they present solutions with $\theta = 10^{-4}$ and 10^{-6} and conclude that such magnitudes have negligible effect. The current study also showed no influence to be present with θ chosen to be a small number.

However, for larger theta, (cases 7 and 8) say θ=0.01 or 0.1, the performance was actually improved in the sense that marching continued farther to larger X_{max}. It should be noted that improvement was noted only for high Reynolds number and small boundary layer cases; namely, the thin subsonic sublayer case. Appreciably viscous flows were unchanged.

10.1 AREAS FOR FUTURE RESEARCH

Three suggestions for possible further research into the algorithm include the use of a sublayer assumption, a nonuniform stepsize and more consistent initial conditions. The scheme should be explored with appropriate pressure assumptions inserted during the derivation of the block tridiagonal form. This will allow investigation of the performance of this scheme with the classical sublayer assumption.

As mentioned earlier, the algorithm of Bhutta and Lewis uses a variable step size based on several factors. In the very viscous cases where the stable step size regime is small,[13] perhaps a correctly calculated step size would provide a more satisfactory marching performance. As the boundary layer and subsonic layers grow in the streamwise direction, the step size can grow accordingly. This is evidenced in Reference 1, but there was no clearly stated methodology to indicate how the step size was calculated or how it varied.

Figure 10.1 shows the constraint schematically. As the subsonic layer increases from level 1 to 2, the minimum step size also increases. Assuming a maximum step size for a given level of accuracy, there is a finite step size window which decreases as the subsonic height increases until there is no stable step size available.

Obtaining a better initial condition could expand on the results of this study. Bhutta and Lewis used a starting code whereas this work uses only an estimate for initial conditions. Indications found during computer runs for Chapter 9 showed that initial conditions which better satisfy the governing equations may allow the solution to march farther, all other factors being equal. The computer runs also showed that the initial profiles for low and high Mach numbers are not necessarily similar.

The initial conditions used in this work are a compromise between Mach numbers 3 and 15. Different initial conditions at each Mach number would have satisfied the governing equations more completely but this was not deemed to be crucial. As an illustration, return to Table 9.1, case 10. The initial conditions (Fig. 6.2) provided for a slightly nonlinear profile slope from the wall to the top of the boundary layer. When this was changed to a linear slope, Case 10 was run for 200 steps at $\Delta x=0.06$ to a final X position of 13.0. The results are shown in Figures 10.2 through 10.5. Judging from the

small right hand side magnitudes of Figure 10.4, the solution was still satisfying the governing equations and would have marched farther. A more consistent set of initial conditions would not change the stability characteristics of the scheme but they would remove an extra source of error, which would aid in marching the solution downstream.

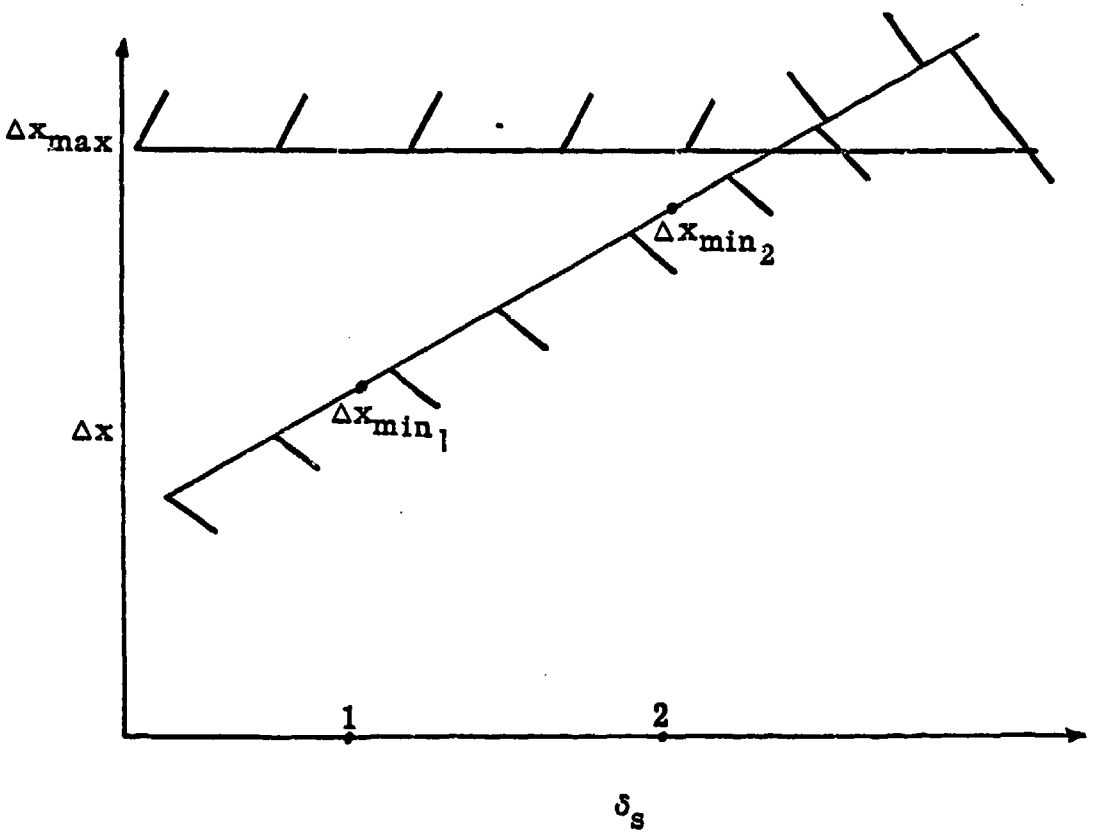


Figure 10.1 Streamwise Stepsize Window

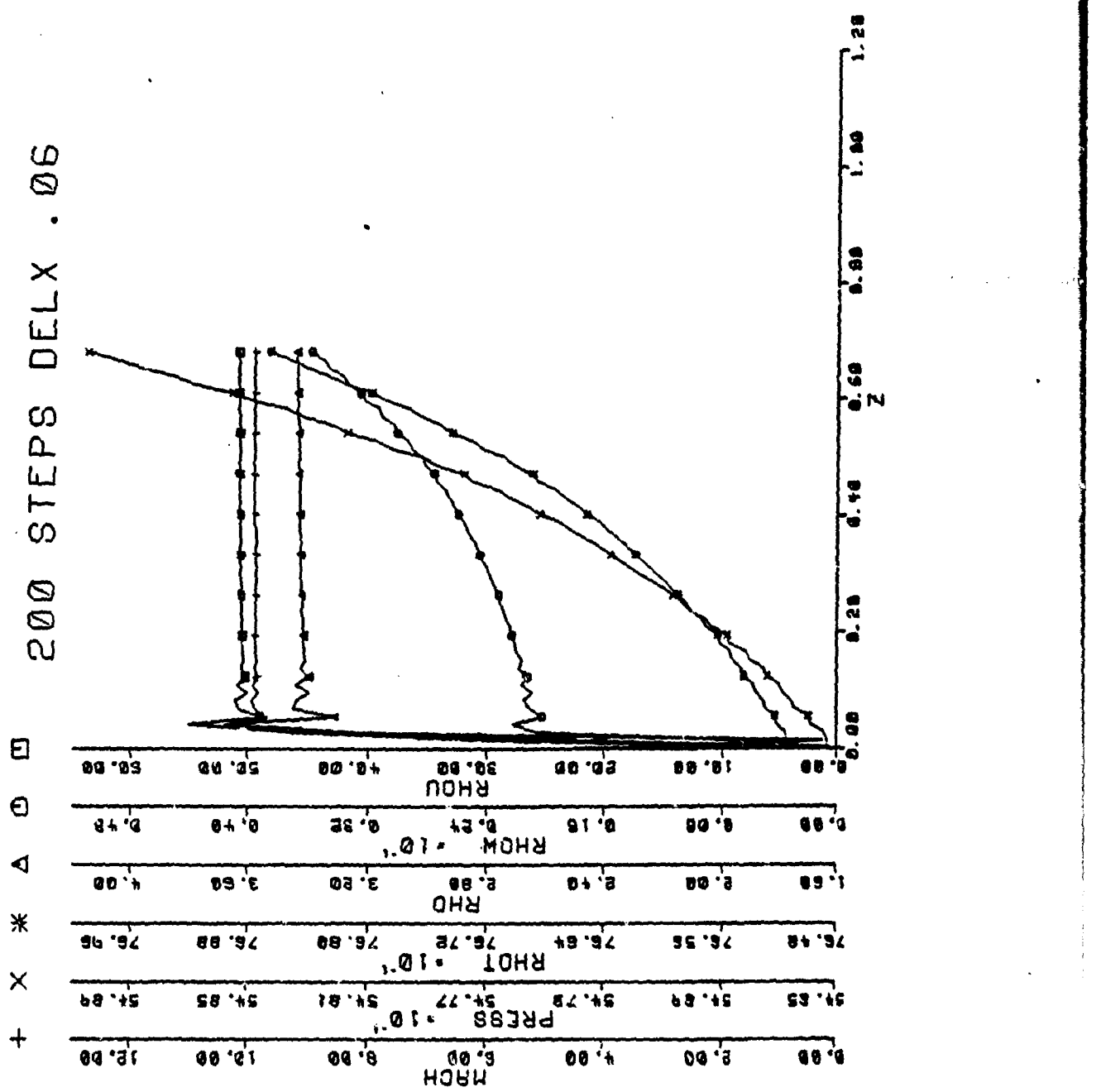


Figure 10.2 State Variable Profiles, X = 13.0

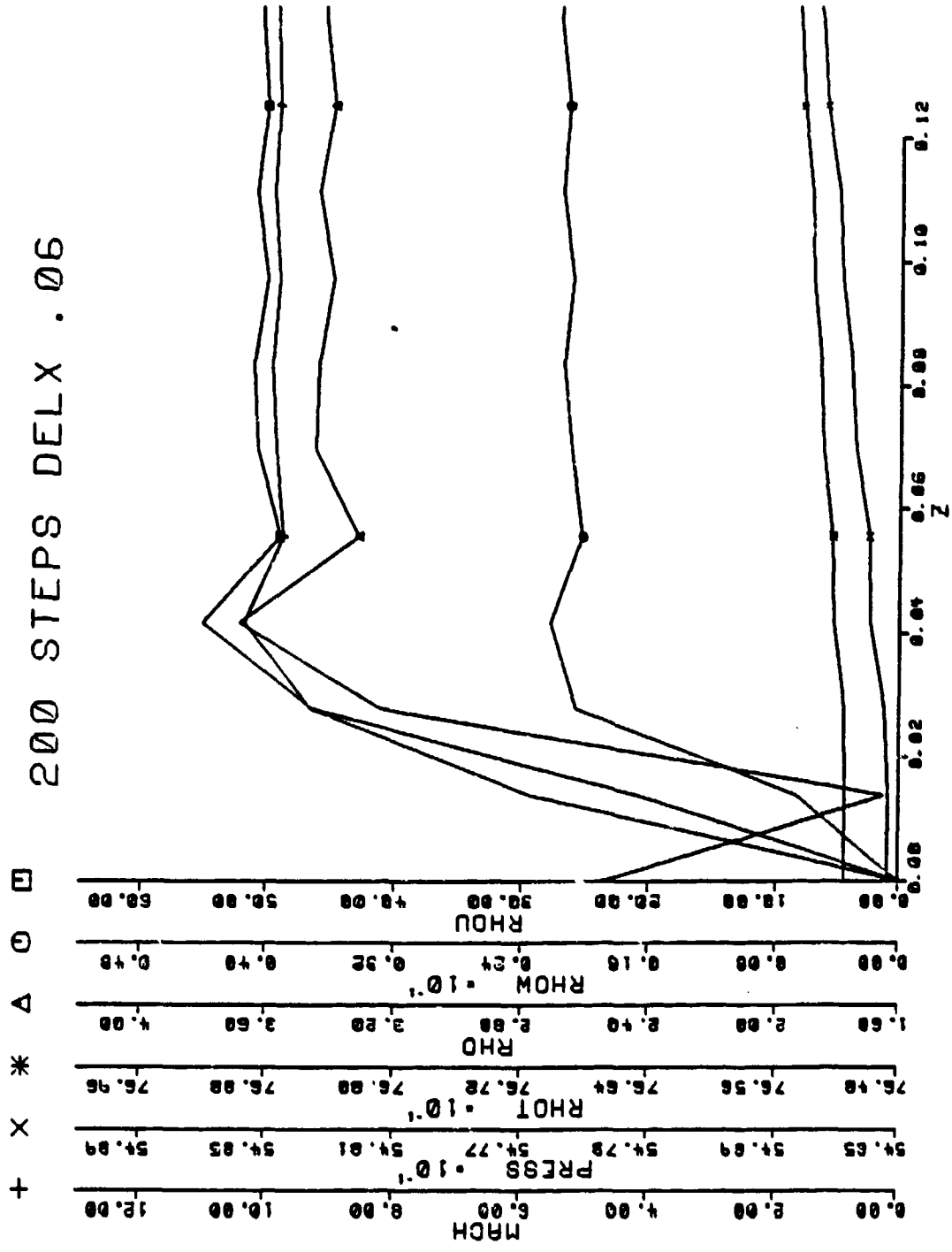


Figure 10.3 Boundary Layer Profiles, X = 13.0

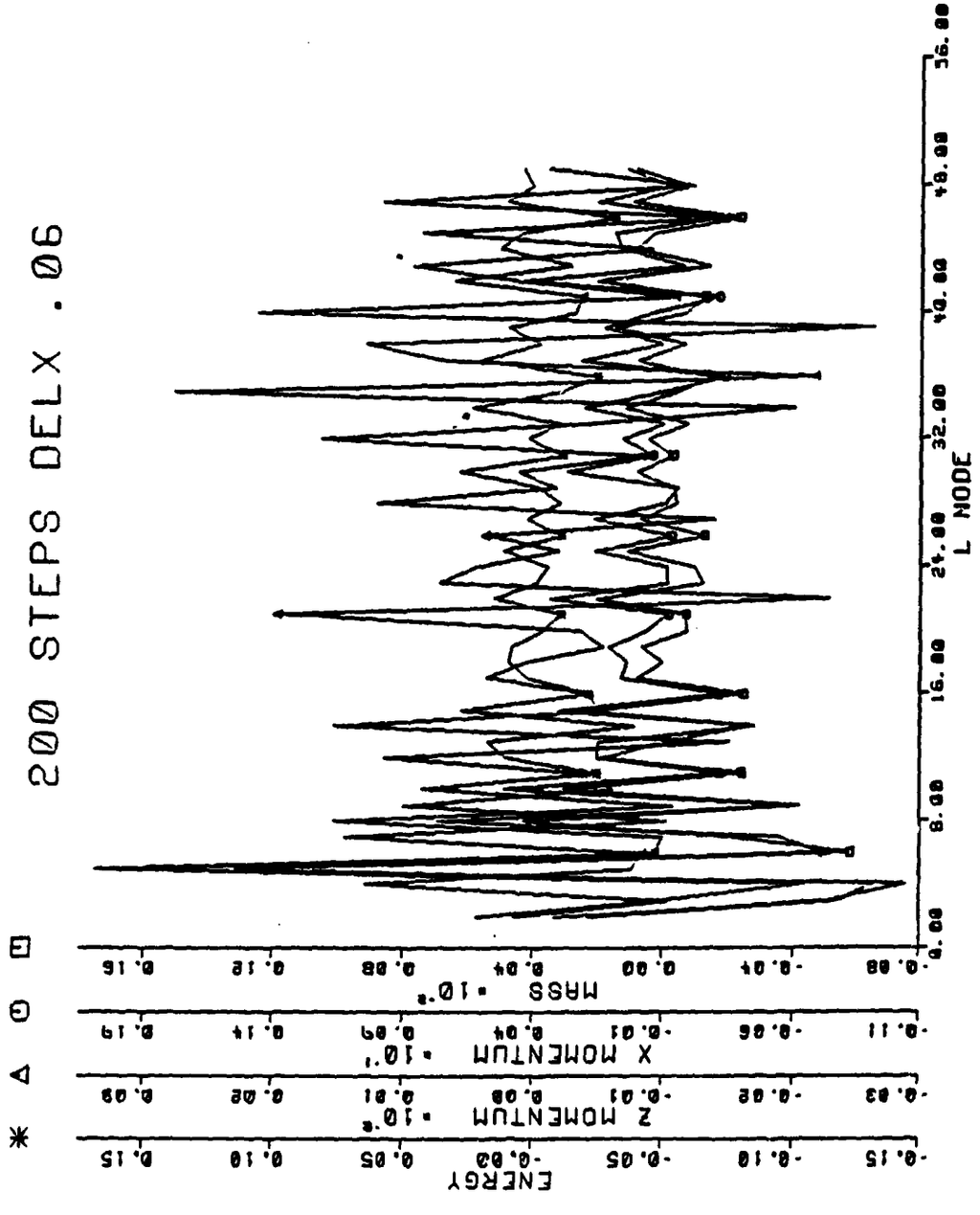


Figure 10.4 Values of Governing Equations, X = 13.0

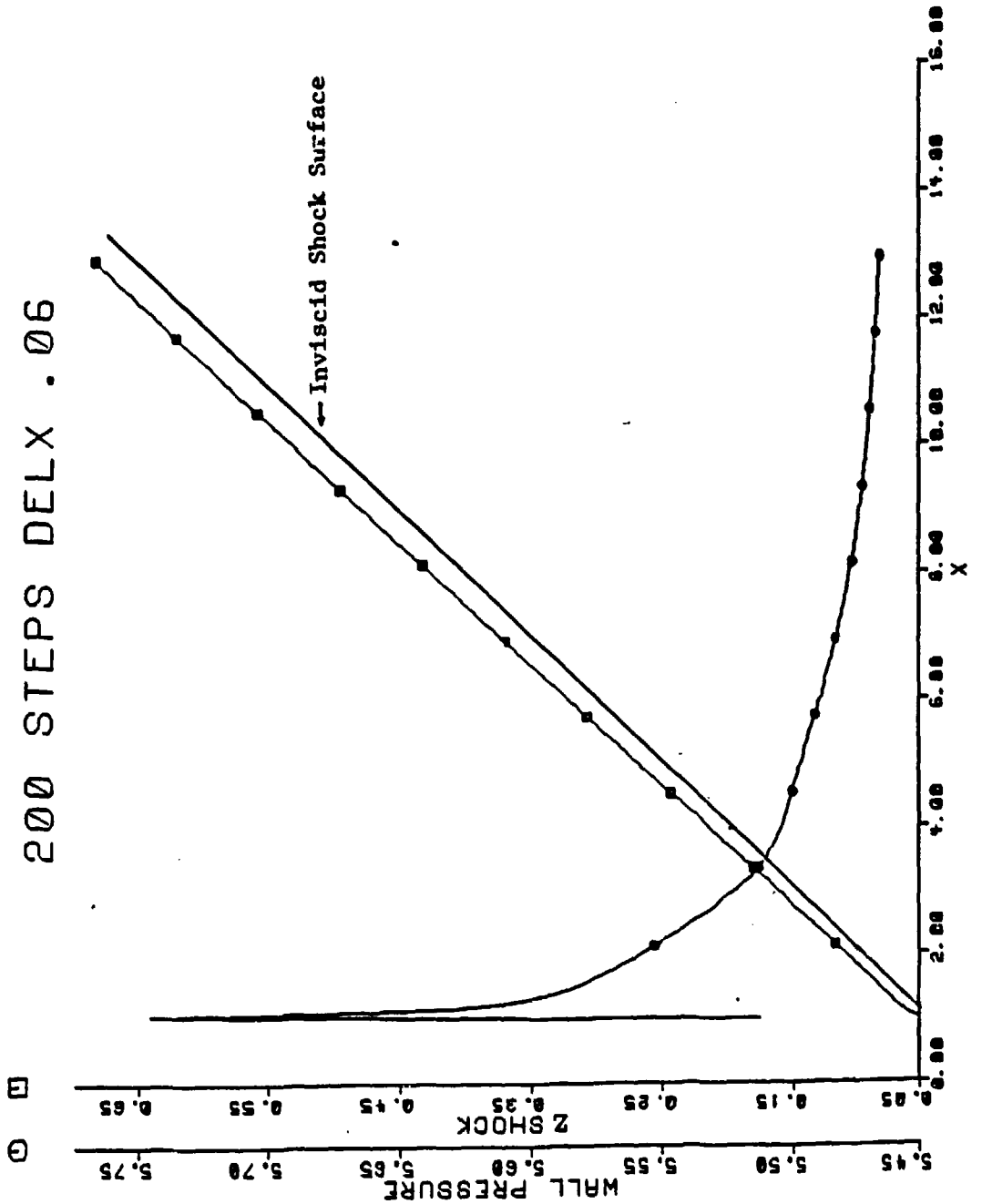


Figure 10.5 Wall Pressure and Shock Surface, X = 13.0

REFERENCES

1. B. A. Bhutta and C. H. Lewis, AIAA 85-0036 "An Implicit Parabolized Navier-Stokes Scheme for High Altitude Reentry Flows", American Institute of Aeronautics and Astronauics, 1985.

2. John D. Anderson, Jr., Introduction to Flight, McGraw-Hill, Inc., 1978.

3. Dale A. Anderson, John C. Tannehill, Richard H. Pletcher, Computational Fluid Mechanics and Heat Transfer, McGraw-Hill, Inc., 1984.

4. Walter J. Krawczyk, Jr., "Evaluation of Supersonic Viscous Three Dimensional Flowfields with Fully Implicit Boundary Conditions", SM Thesis, Massachusetts Institute of Technology, 1983.

5. Ovid W. Eshbach, Editor, Handbook of Engineering Fundamentals, John Wiley & Sons, 1975.

6. Professor Michael B. Giles, "B3SOLV", Massachusetts Institute of Technology, 1985.

7. D.S. Chausee, J.L. Patterson, P. Kutler, T.H. Pulliam, J.L. Steger, AIAA Paper 81-0050 "A Numerical Simulation of Hypersonic Viscous Flows over Arbitrary Geometries at High Angle of Attack, American Institute of Aeronautics and Astronautics, 1981.

8. Lewis B. Schiff and Joseph L. Steger, "Numerical Simulation of Steady Supersonic Viscous Flows", NASA Technical Paper 1749, National Aeronautics and Space Administration, 1981.

9. M. Barnett, "The Solution of the Parabolized Navier-Stokes Equations by a Fully Implicit Method", AIAA Paper 82-0415, American Institute of Aeronautics and Astronautics, 1982.

10. James R. Shrock, MULTIGRAPH, MIT Lincoln Laboratory, 1986.

11. B.A. Bhutta and C.H. Lewis, AIAA 85-1604 "Prediction of Three-Dimensional Hypersonic Reentry Flows Using a PNS Scheme", American Institute of Aeronautics and Astronautics, 1985.

12. Roger Peyret and Henri Viviand, "Computation of Viscous Compressible Flows Based on the Navier-Stokes Equations", AGARD-AG-212, NATO Advisory Group for Aerospace Research and Development, 1975.

13. S.G. Rubin and A. Lin, "Marching with the Parabolized Navier-Stokes Equations", Isreal Journal of Technology, Volume 18, 1980.

14. Upender K. Kaul, "A Relaxation Technique for the Parabolized Navier-Stokes (PNS) Equations", Communications in Applied Numerical Methods, Volume 2, 1986.

15. R.T. Davis, M. Barnett, and J.V. Rakich, "The Calculation of Supersonic Viscous Flows Using the Parabolized Navier-Stokes Equations", Computers and Fluids, Volume 14, No. 3, 1986.

APPENDIX A

EQUATION TRANSFORMATION DERIVATION

Most of the basic transformation from a rectangular to a general coordinate system was performed in Chapter 3. The transformed equations are 3.13a and 3.14. The final step is to transform the individual components of the vectors in the governing equations.

Comparing 3.4, 3.13a, and 3.14 we see that:

$$F_1 = \xi_t E_i + \xi_x G_i \qquad A.1$$
$$F_2 = \zeta_t E_i + \zeta_x G_i$$

Substituting for Ei and Gi above, the following forms result for F_1 and F_2:

$$F_1 = 1/J \begin{pmatrix} \rho(\xi_t u + \xi_x w) \\ \rho u(\xi_t u + \xi_x w) + \xi_x P \\ \rho w(\xi_t u + \xi_x w) + \xi_x P \\ (T/(\gamma-1) + V^2/2)\rho(\xi_t u + \xi_x w) \\ 0 \end{pmatrix} \qquad A.2$$

$$F_2 = 1/J \begin{pmatrix} \rho(\zeta_t u + \zeta_x w) \\ \rho u(\zeta_t u + \zeta_x w) + \zeta_x P \\ \rho w(\zeta_t u + \zeta_x w) + \zeta_x P \\ (T/(\gamma-1) + V^2/2)\rho(\zeta_t u + \zeta_x w) \\ 0 \end{pmatrix} \qquad A.3$$

If the modified differential state equation is used, the final component of F_1 and F_2 will be θp. The 1/J factor is added to preserve the strong conservation law form.[3] It is differenced with the other components of the vector which preserves the conservation property. Note that in Reference 1 the contravariant velocities:

$$(\xi_x u + \xi_z w) \text{ and } (\zeta_x u + \zeta_z w)$$

were printed incorrectly.

The last vector to be transformed is the viscous vector since the vector H is algebraic and is not transformed. The viscous vector is slightly more complicated than F_1 and F_2 because it contains X and Z derivatives which must also be transformed. From Equations 3.4, 3.13a, and 3.14 again:

$$S = \xi_x Ev + \zeta_x Gv$$

where Gv was seen in equation 3.4 and

$$Ev = \begin{pmatrix} 0 \\ \tau_{xx} \\ \tau_{zx} \\ u\tau_{xx} + w\tau_{zx} - q_x \\ 0 \end{pmatrix}$$

The derivation will be carried out component by component.

The first component of S is zero. The second component is:

$$\xi_x \tau_{x x} + \xi_z \tau_{z x}$$

which is expanded as:[3]

$$\mu[\xi_x (4u_{,x} - 2w_{,z})/3 + \xi_z (w_{,x} + u_{,z})]$$

These X,Z derivatives must also be transformed to the ξ, ζ system.

Using 3.12 the above expansion is transformed to:

$$\mu(\xi_x [4(\xi_x u_{,\xi} + \xi_x u_{,\zeta})/3 - 2(\xi_x w_{,\xi} + \xi_x w_{,\zeta})/3] + \xi_z [\xi_x w_{,\xi} + \xi_x w_{,\zeta} + \xi_x u_{,\xi} + \xi_x u_{,\zeta}]) \qquad \text{A.4}$$

Following the PNS assumptions, all streamwise viscous derivatives are dropped from A.4 leaving:

$$(\mu/J)[M_0 u_{,\zeta} + (M_{x x} u_{,\zeta} + M_{x z} w_{,\zeta})/3] \qquad \text{A.5}$$

where

$$M_0 = \xi_z^2 + \xi_z^2$$

$$M_{x x} = \xi_x^2$$

$$M_{x z} = \xi_x \xi_z$$

The third component:

$$\xi_x \tau_{x z} + \xi_z \tau_{z z}$$

is done in the same manner, resulting in:

$$(\mu/J)[M_0 w_{,\zeta} + (M_{x z} u_{,\zeta} + M_{z z} w_{,\zeta})/3] \qquad \text{A.6}$$

where

$$M_{t_x} = \xi_x^2$$

The fourth component is:

$$\xi_x(u\tau_{x_x} + v\tau_{x_y} - q_x) + \xi_y(u\tau_{x_y} + v\tau_{x_z} - q_y)$$

where

$$q_x = -k(T,_x) \qquad q_y = -k(T,_y)$$

The final form of the entire viscous vector is given by:

$$S = \mu/J \begin{pmatrix} 0 \\ [M_\infty u,_\zeta + (M_{x_x} u,_\zeta + M_{x_y} v,_\zeta)/3] \\ [M_\infty v,_\zeta + (M_{x_x} u,_\zeta + M_{x_y} v,_\zeta)/3] \\ [M_\infty (T,_\zeta/(Pr(\gamma-1)) + uu,_\zeta + vv,_\zeta) \\ \quad + (M_{x_x} uu,_\zeta + M_{x_y} vv,_\zeta + M_{x_x}(vu,_\zeta + uv,_\zeta))/3] \\ 0 \end{pmatrix}$$

In Reference 1, the first two of the 1/3 coefficients of S were incorrectly printed as 1/2.

APPENDIX B

JACOBIAN MATRICES

After the coordinate transformation, F_1, F_2, S, and H are given by 3.15 ,3.16 and Appendix A. The state vector after the same transformation is

$$q = 1/J\ [\rho, \rho u, \rho w, \rho T, p]^T \qquad B.1$$

The Jacobian matrices are formed by the following partial derivatives:

$$A_0 = H,q \quad A_1 = F_1,q \quad A_2 = F_2,q \quad M = S,q \qquad B.2$$

For example, the first row, second column of Matrix A_1 is the partial derivative of the first component of F_1 with respect to the second component of q (Eq. B.1). The fourth row, fourth column of matrix A_2 is the partial derivative of the fourth component of F_2 with respect to the fourth component of q (Eq. B.1). Each component of q is treated as one term when taking derivatives. They are not broken down into primitive variables.

Following are the four Jacobian matrices using the modified equation of state.

$$A_0 = J \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & \gamma \end{bmatrix}$$

$$A_1 = \begin{bmatrix} 0 & \xi_x & \xi_z & 0 & 0 \\ -uU_1 & 2\xi_x + \xi_z w & \xi_z u & 0 & \xi_x \\ -wU_1 & \xi_x w & \xi_x u + 2\xi_z w & 0 & \xi_z \\ T_1 & T_2 + \xi_z uw & T_3 + \xi_x uw & U_1/(\gamma-1) & 0 \\ 0 & 0 & 0 & 0 & \theta \end{bmatrix}$$

where

$$U_1 = \xi_x u + \xi_z w$$

$$T_1 = -U_1 (T/(\gamma-1) + V^2)$$

$$T_2 = \xi_x (T/(\gamma-1) + (3u^2 + w^2)/2)$$

$$T_3 = \xi_z (T/(\gamma-1) + (u^2 + 3w^2)/2)$$

$$A_2 = \begin{bmatrix} 0 & \zeta_x & \zeta_z & 0 & 0 \\ -uU_2 & 2\zeta_x + \zeta_z w & \zeta_z u & 0 & \zeta_x \\ -wU_2 & \zeta_x w & \zeta_x u + 2\zeta_z w & 0 & \zeta_z \\ T_4 & T_5 + \zeta_z uw & T_6 + \zeta_x uw & U_1/(\gamma-1) & 0 \\ 0 & 0 & 0 & 0 & \theta \end{bmatrix}$$

where

$$U_2 = \zeta_x u + \zeta_z w$$

$$T_4 = -U_2 (T/(\gamma-1) + V^2)$$

$$T_x = \xi_x \left(T/(\gamma-1)+(3u^2+w^2)/2\right)$$

$$T_z = \xi_z \left(T/(\gamma-1)+(u^2+3w^2)/2\right)$$

$$M = -\mu/J \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ \beta D_1 + \alpha D_2 & -\beta D_3 & -\alpha D_3 & 0 & 0 \\ \sigma D_2 + \alpha D_1 & -\alpha D_3 & -\sigma D_3 & 0 & 0 \\ \{ND_4 + \beta D_5 & -\beta D_1 - \alpha D_2 & -\sigma D_2 - \alpha D_1 & -ND_3 & 0 \\ +\sigma D_6 + 2\alpha D_7\} & & & & \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

where

$$\beta = M_0 + M_{x\,x}/3$$

$$\alpha = M_{z\,x}/3$$

$$\sigma = M_0 + M_{z\,z}/3$$

$$N = M_0/(Pr(\gamma-1))$$

The following are all partial derivatives contained in M:

$$D_1 = (uJ/\rho),_\xi$$

$$D_2 = (wJ/\rho),_\xi$$

$$D_3 = (J/\rho),_\xi$$

$$D_4 = (TJ/\rho),_\xi$$

$$D_5 = (u^2 J/\rho),_\xi$$

$$D_6 = (w^2 J/\rho),_\xi$$

$$D_7 = (uwJ/\rho),_\xi$$

APPENDIX C

MISCELLANEOUS DERIVATIONS

MASS FLOW, U_e, W_e, SHOCK PREDICTION

U_2, W_2

The velocities across the shock are determined from geometrical relationships. Figure C.1 illustrates the nondimensional velocity components in the wedge coordinate system. The shock angle, and therefore the velocities across the shock, are functions of downstream position.

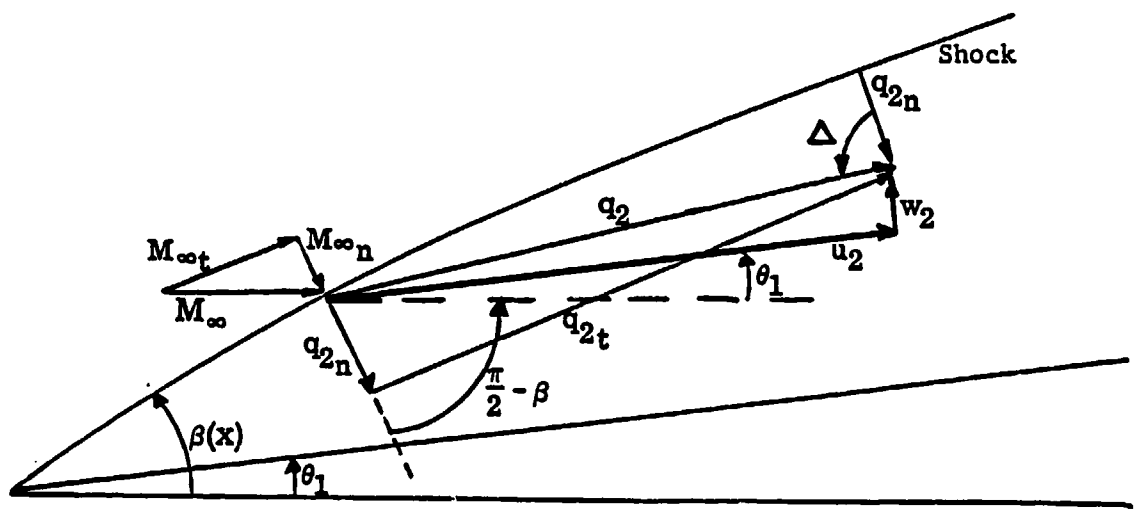


Figure C.1 Shock Velocity Geometry

$$\Delta = \tan^{-1}(q_{2_t}/q_{2_n})$$

$$= [\tan^{-1}(W_2/U_2) + \pi/2 - \beta + \theta_1] \qquad C.1$$

and

$$q_{2_n} = q_2 \cos[\Delta]$$

$$= (U_2^2 + W_2^2)^{1/2}(\cos\Delta) \qquad C.2$$

Equations C.1 and C.2 are two equations in the two unknowns, U_2 and W_2. Solving simultaneously gives:

$$AA = 1 + \tan^2(\beta - \pi/2 - \theta_1 + \cos^{-1}(\tan\beta/(\rho_2^2 + \tan^2\beta)^{1/2})$$

$$U_2 = [M_{\infty_n}^2(1/\tan^2\beta + 1/\rho_2^2)/AA]^{1/2} \qquad C.3$$

$$W_2 = U_2(AA) \qquad C.4$$

MASS FLOW

The geometric relationships for mass flow are shown in Figure C.2. The freestream normal distance, Z_∞', must be found in terms of the wedge Z coordinate.

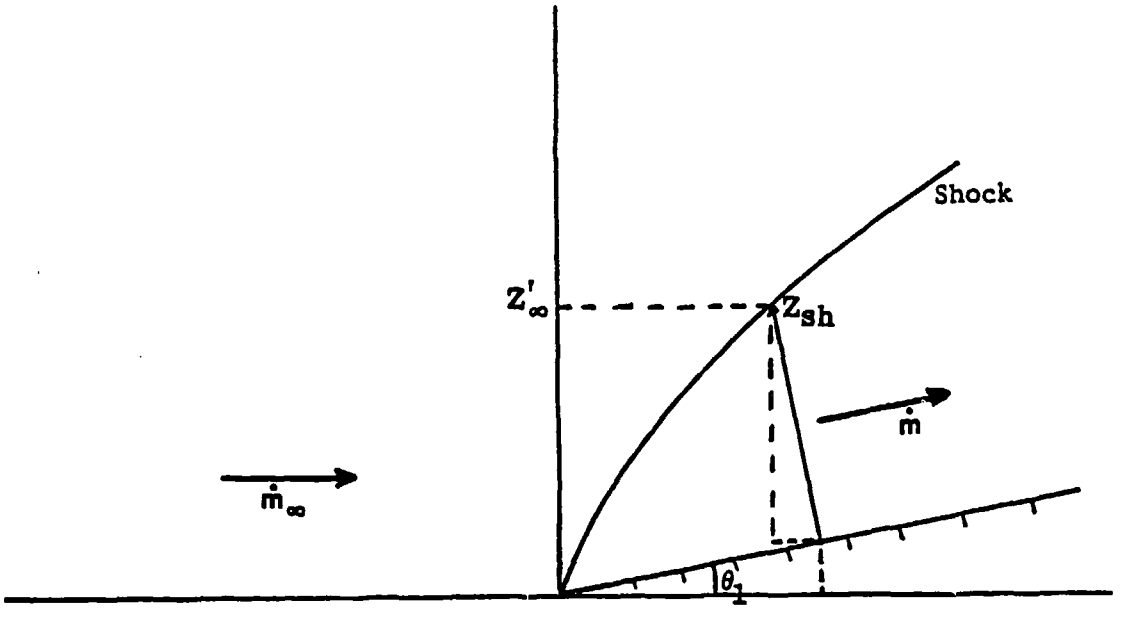


Figure C.2 Mass Flow Geometry

For mass flow to balance, it must satisfy:

$$(\rho u Z)_\infty = \int_0^{z_{SHOCK}} \rho u dz \qquad C.5$$

After nondimensionalizing and finding the relationship for Z_∞, Equation C.5 becomes:

$$M_\infty(Z_{s_N}\cos(\theta_1) + X\sin(\theta_1)) = \int_0^{z_{SHOCK}} \rho u dz \qquad C.6$$

To discretize the integral in C.6, the vertical nodes (L) of Chapter 5 are used. The integral is then:

$$\Sigma_L (\rho u)\Delta z \qquad C.7$$

where the summation is carried out over the vertical nodes.

The value of the mass flow across the shock (Equ. C.7) normalized with respect to the freestream mass flow (Equ. C.6) should be near 1.0. If not, the shock position is moved up or down to equalize the two mass flows. Reference 1 suggests that a difference of 0.1% is sufficient.

SHOCK PREDICTION

From the reference by Chausee et al.[7], the shock location in two dimensions can be predicted from location j to $j+1$. Differences in coordinate systems have been accounted for. The next shock location is given by:

$$Z^{j+1} = Z^j + Z,_\xi \Delta\xi \qquad C.8$$

where

$$Z,_\xi = (\eta_y \xi_x - \eta_x \xi_x)/J$$

which in two dimensions reduces to:

$$M_{1_n}/(M_\infty^2 - M_{1_n}^2)^{1/2} = M_{1_n}/M_{1_t} = \tan\beta \qquad C.9$$

(Fig C.1). Then Equation C.8 becomes

$$Z^{j+1} = Z^{j} + \tan\beta\Delta x$$

This is shown geometrically in Figure C.3

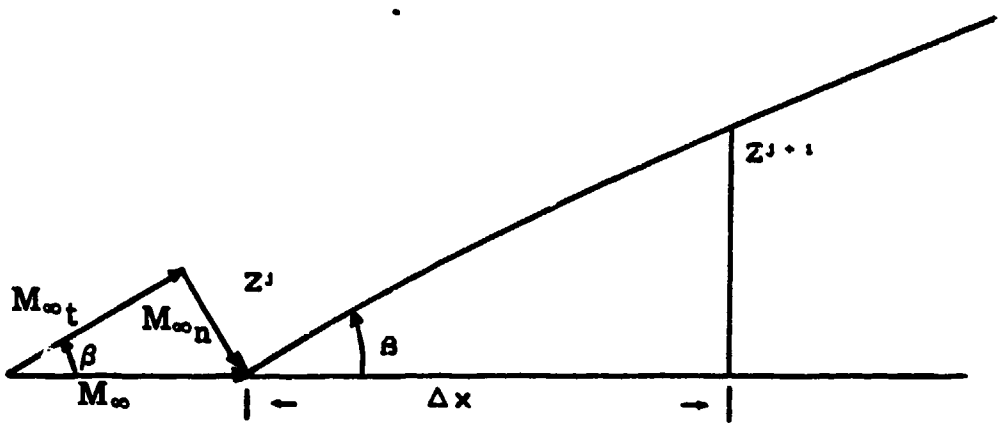


Figure C.3 Shock Prediction

APPENDIX D

COMPUTER CODE

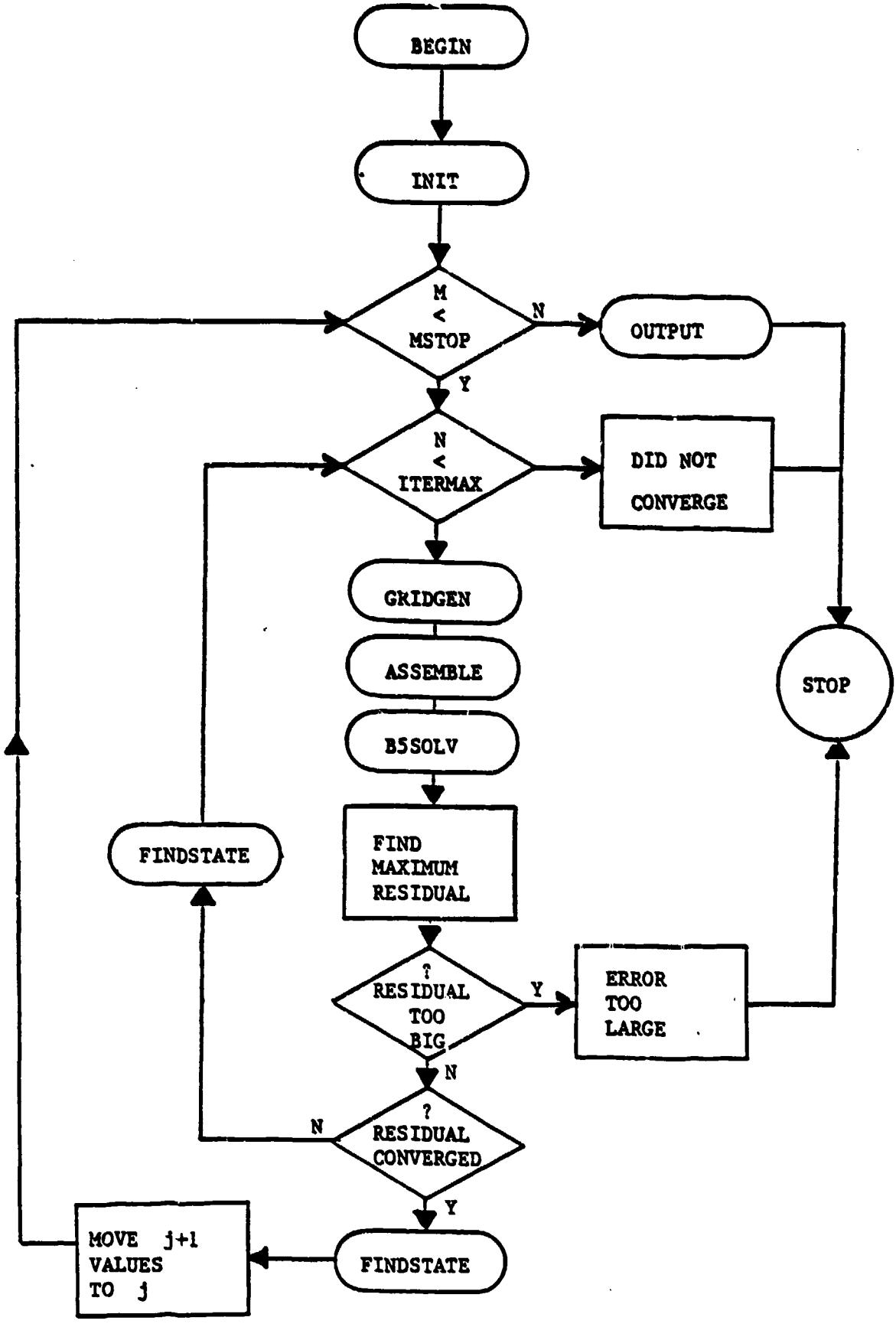


Figure D.1 Flowchart

The code is titled PARABNS.FOR and uses the file PARABNS.INC. Both are written in FORTRAN and have been run on a Digital VAX 11/780 and a MicroVAX.

SUBROUTINES

INIT - Takes input of the boundary layer thickness and makes estimates for the initial profiles. Uses massflow to refine initial estimate for shock angle. Inviscid shock angle and Rankine-Hugoniot relations provide initial estimates at the shock.

GRIDGEN - If the time level is at the first iteration, the routine predicts the j+1, n=1 state vector based on past state vectors. Also predicts new shock location. If the time level is greater than one, mass flow is used to refine the shock location. Interior values of density are used to predict the shock density and therefore the shock angle. Other shock values are found from the Rankine-Hugoniot relations. Metrics and X,Z positions are computed.

ASSEMBLE - Assembles the individual Jacobian matrices into block tridiagonal form. It also performs the tridiagonal boundary conditions. Two subroutines are called:

A) FINDG - Calculates the right hand side vector (Equ. 5.8).

B) BUILDMATRIX - Fills the individual Jacobian

matrices: A_0, A_1, A_2 and M

BSSOLV - 5 x 5 Block tridiagonal solver

FINDSTATE - Smoothes the $\Delta\Omega^{n+1}$ terms and updates the state vector with the newly smoothed solution.

OUTPUT - Creates 5 output files:

A) State Variable Profiles

B) Convergence History

C) Governing Equation Errors

D) Shock Surface and Wall Pressure

E) Streamwise Profiles of Selected State Variables

A copy of the computer code is on file with Professor Judson R. Baron, Massachusetts Institue of Technology, Room 33-217, (617) 253-4329.